SUBJECT: Deep Learning and Computer Vision
SUBJECT CODE: ECS795P

NAME: Ganesh Kumaran Masilamani
STUDENT NO: 200434339

APPENDIX – A : Mini Project Architectures and Outputs

A.1 Figures and Tables

I have created 5 Models which includes **VGG16, ResNet, GoogleNet, AlexNet** and **MobileNet** from the scratch (i.e, coded layer by layer to form the Network Architecture) for this Mini-project as my Main Files. The developed models has been trained and evaluated the developed models, also I have visualized the predicted outputs and the Individual accuracy of all the individual class in the used dataset **(MNIST and CIFAR10)**.

Main Python Notebook File Names which is Model Architecture is built from Scratch,

Files which uses MNIST dataset

- MiniProject_VGG_MNIST_model_200434339_Ganesh_Kumaran_Masilamani.ipynb
- MiniProject_ResNet_MNIST_model_200434339_Ganesh_Kumaran_Masilamani.ipynb
- MiniProject_GoogleNet_MNIST_model_200434339_Ganesh_Kumaran_Masilamani.ipynb
- MiniProject_AlexNet_MNIST_model_200434339_Ganesh_Kumaran_Masilamani.ipynb
- MiniProject_MobileNet_MNIST_model_200434339_Ganesh_Kumaran_Masilamani.ipynb

Files which uses CIFAR10 dataset

- MiniProject_VGG_Cifar10_model_200434339_Ganesh_Kumaran_Masilamani.ipynb
- MiniProject_ResNet_Cifar10_model_200434339_Ganesh_Kumaran_Masilamani.ipynb
- MiniProject_GoogleNet_Cifar10_model_200434339_Ganesh_Kumaran_Masilamani.ipynb

A.1.1 VGG16 Architecture

We build our VGG16 Model with the sequence of the layers, Input dimensions are fixed which is 244×244 image size. The first layer is the Convolutional layer which is the 3×3 convolutional layer with padding, this is to maintain the resolution in the model. The second layer will be the ReLU function which is used for the activation which is often tends to be Non-linear and the final layer would be the Maxpooling layer which is used for downsampling. Also with the value of 0.2 in the dropout layer.

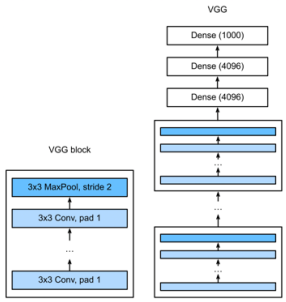


Fig 1. Architecture of VGG16

A.1.2 Residual Block Architecture for ResNet

The first layer of the residual block is 3×3 convolutional layer and then the data needs to be normalized so that, it is passed through the Batch-normalization.

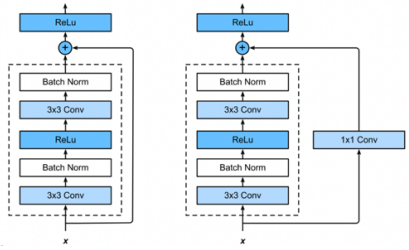


Fig 2. Architecture of Residual Block

A.1.3 ResNet Architecture

ResNet is constructed using the 4 Residual Blocks in which Network skips some layers. The Residual block is connected with 7×7 Convolutional Layer, 3×3 Maxpooling layer and finally it is passed through the Global average pooling layer in which helps the data not getting overfitted by enforcing correspondences among mapping and the features .

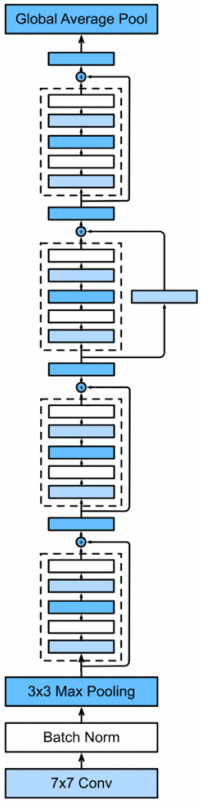


Fig 3. Architecture of ResNet

A.1.4 Inception Block Architecture for GoogleNet

The Inception Block consists of 4 layer in which the first layer has the input connected with the 1×1 Convolutional network. The second layer has the input connected with the 1×1 convolutional network and then it is passed through the 3×3 convolutional network. When coming to the third layer same happens like the second layer, but instead of 3×3 it is connected to the 5×5 convolutional Network, and the final layer consists of 3×3 Max pooling layer and the 1×1 convolutional network. Finally all the four layers are concatenated, this forms the inception block.

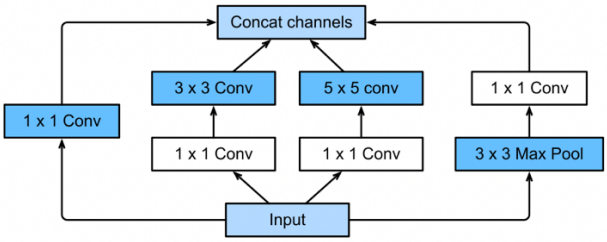


Fig 4. Architecture of Inception Block

A.1.5 GoogleNet Architecture

Dataset is passed through the 7×7 convolutional layer and passed through the 3×3 Max pooling layer and transferred through the 1×1 and 3×3 convolutional layer and as per the GoogleNet architecture it is passed through the 3×3 max pooling layer. Next step is to pass the inception block two times in the network and again passed with 3×3 maxpooling layer. Then it is passed with the Inception block 5 times and with Maxpooling layer finally Inception block is passed two times and finally it is passed with the global average pooling to make all the layers fully connected and forms the Network.

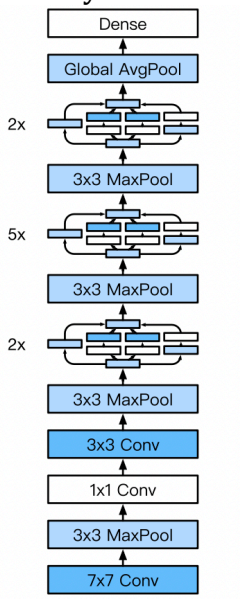


Fig 5. Architecture of GoogleNet

A.1.6 AlexNet Architecture

Alexnet is built with the first layer containing 11×11 convolutional layer with maxpooling in the first convolution window, the second and third convolution window has 5×5 and three layers of 3×3 convolutional layers with Maxpooling layer. Finally network is fully connected using the Dense layer which contains 4096 outputs.

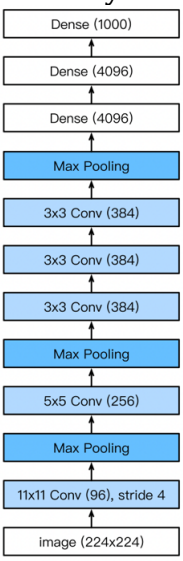


Fig 6. Architecture of AlexNet

A.1.7 MobileNet Architecture

MobileNet architecture uses depthwise convolutions to construct a deep network. I consists of 28layers which has both depthwise seperable convolutional layers and the pointwise seperable convolutional layers. Depthwise seperable convolution layer is formed by the combination of the depthwise and pointwise convolution. Depthwise convolution is maps a single input at a time in the input channel. The dimension of the input channel is equal to the dimension of the output channel. The pointwise convolution combines the features which was produced in the depthwise convolution. This gives raise to the new features. Finally global average pooling is used to form the fully connected network.

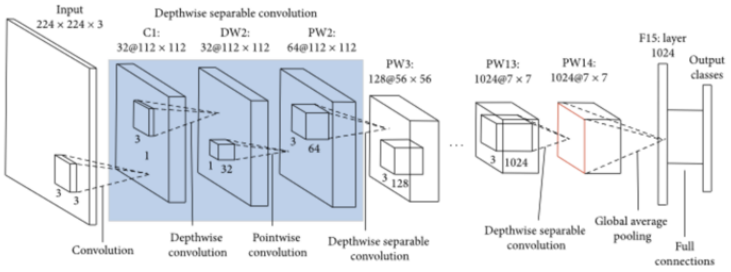


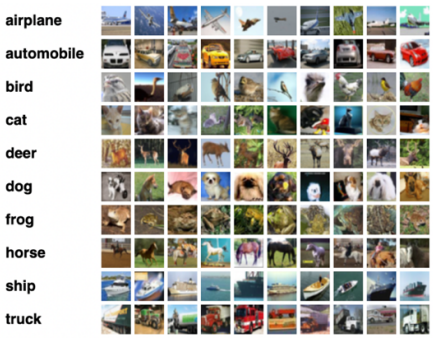
Fig 7. Architecture of MobileNet

A.1.8 Example of MNIST dataset

MNIST dataset which contains 60000 samples splitted into 50000 training samples and the 10000 test samples, which is of 28×28 Image pixel size. This dataset is available in the MNIST database website.

Fig 8. MNIST Dataset

A.1.9 Example of CIFAR10 dataset



CIFAR10 dataset is used for the further investigation of the models. CIFAR10 dataset is available in the Toronto.edu site which contains 60000 images separated among 10 classes of 32×32 images size.

Fig 9. CIFAR10 Dataset

A.1.10 VGG16 Architecture Results on MNIST data

```
------------------------------------------------------
Model Training and Testing Start date: 14/05/2021
Model Training and Testing starting time: 23:15:40
------------------------------------------------------
VGG16 Architecture Training and Testing Begins!
-------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 1      Train_Loss: 0.0250      Train_Accuracy: 70.82      Test_Loss: 0.0015      Test_Accuracy: 98.54     Time_Duration: 258.1 sec
-------------------------------------------------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 2      Train_Loss: 0.0015      Train_Accuracy: 98.62      Test_Loss: 0.0009      Test_Accuracy: 99.06     Time_Duration: 258.0 sec
-------------------------------------------------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 3      Train_Loss: 0.0009      Train_Accuracy: 99.10      Test_Loss: 0.0009      Test_Accuracy: 99.06     Time_Duration: 258.0 sec
-------------------------------------------------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 4      Train_Loss: 0.0006      Train_Accuracy: 99.40      Test_Loss: 0.0007      Test_Accuracy: 99.34     Time_Duration: 257.8 sec
-------------------------------------------------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 5      Train_Loss: 0.0004      Train_Accuracy: 99.57      Test_Loss: 0.0007      Test_Accuracy: 99.29     Time_Duration: 257.9 sec
-------------------------------------------------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 6      Train_Loss: 0.0003      Train_Accuracy: 99.74      Test_Loss: 0.0006      Test_Accuracy: 99.51     Time_Duration: 257.8 sec
-------------------------------------------------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 7      Train_Loss: 0.0002      Train_Accuracy: 99.81      Test_Loss: 0.0008      Test_Accuracy: 99.32     Time_Duration: 257.9 sec
-------------------------------------------------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 8      Train_Loss: 0.0002      Train_Accuracy: 99.78      Test_Loss: 0.0007      Test_Accuracy: 99.34     Time_Duration: 257.9 sec
-------------------------------------------------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 9      Train_Loss: 0.0002      Train_Accuracy: 99.79      Test_Loss: 0.0007      Test_Accuracy: 99.34     Time_Duration: 257.8 sec
-------------------------------------------------------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 10     Train_Loss: 0.0001      Train_Accuracy: 99.86      Test_Loss: 0.0007      Test_Accuracy: 99.25     Time_Duration: 257.8 sec
-------------------------------------------------------------------------------------------------------------------------------------------------------
VGG16 Architecture Training and Testing Completed!
------------------------------------------------------
Model Training and Testing End date: 14/05/2021
Model Training and Testing End time: 23:58:54
------------------------------------------------------
```

Fig 10. VGG16 Results on MNIST data

A.1.11 Original and Predicted image classes for MNIST on VGG16

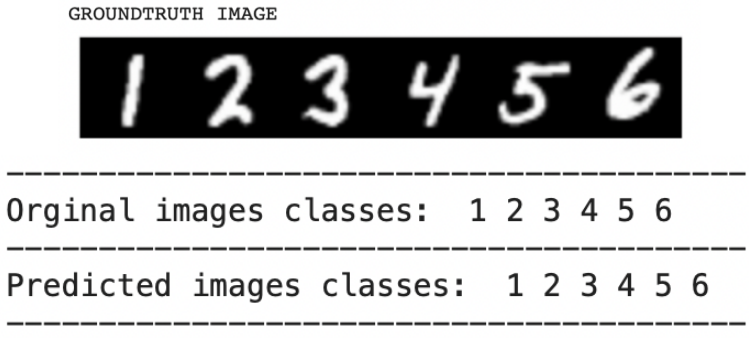
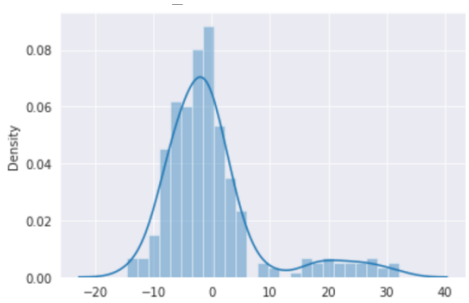


Fig 11. Original and Predicted image classes for MNIST on VGG16

Note: Output plot of the final tensor values for MNIST data on VGG16



A.1.12 Individual accuracies of the image classes for MNIST in VGG16

```
--------------------------
Accuracy of     0 : 99 %
--------------------------
--------------------------
Accuracy of     1 : 99 %
--------------------------
--------------------------
Accuracy of     2 : 99 %
--------------------------
--------------------------
Accuracy of     3 : 98 %
--------------------------
--------------------------
Accuracy of     4 : 99 %
--------------------------
--------------------------
Accuracy of     5 : 98 %
--------------------------
--------------------------
Accuracy of     6 : 99 %
--------------------------
--------------------------
Accuracy of     7 : 99 %
--------------------------
--------------------------
Accuracy of     8 : 99 %
--------------------------
--------------------------
Accuracy of     9 : 99 %
--------------------------
```

Fig 12. Individual accuracies of the image classes for MNIST in VGG16

Note: Wrong Predictions on MNIST data(Failure)

GROUNDTRUTH IMAGE

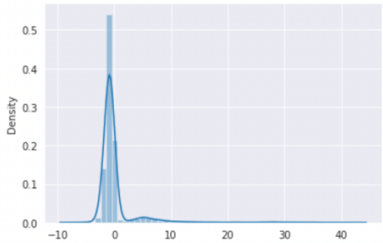
```
-----------------------------------------
Orginal images classes:  1 2 3 4 5 6
-----------------------------------------
Predicted images classes:  3 3 3 3 3 3
-----------------------------------------
```

A.1.13 ResNet Architecture Results on MNIST data

```
--------------------------------------------------
Model Training and Testing Start date: 14/05/2021
Model Training and Testing starting time: 23:15:46
--------------------------------------------------
ResNet Architecture Training and Testing Begins!
--------------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 1      Train_Loss: 0.0073      Train_Accuracy: 94.31      Test_Loss: 0.0013     Test_Accuracy: 98.67    Time_Duration: 42.7 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 2      Train_Loss: 0.0012      Train_Accuracy: 98.83      Test_Loss: 0.0008     Test_Accuracy: 99.10    Time_Duration: 42.7 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 3      Train_Loss: 0.0008      Train_Accuracy: 99.24      Test_Loss: 0.0009     Test_Accuracy: 99.08    Time_Duration: 42.8 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 4      Train_Loss: 0.0005      Train_Accuracy: 99.50      Test_Loss: 0.0007     Test_Accuracy: 99.32    Time_Duration: 42.8 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 5      Train_Loss: 0.0004      Train_Accuracy: 99.65      Test_Loss: 0.0009     Test_Accuracy: 99.12    Time_Duration: 42.7 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 6      Train_Loss: 0.0003      Train_Accuracy: 99.74      Test_Loss: 0.0012     Test_Accuracy: 98.97    Time_Duration: 42.8 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 7      Train_Loss: 0.0002      Train_Accuracy: 99.79      Test_Loss: 0.0008     Test_Accuracy: 99.29    Time_Duration: 42.8 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 8      Train_Loss: 0.0002      Train_Accuracy: 99.83      Test_Loss: 0.0007     Test_Accuracy: 99.50    Time_Duration: 42.7 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 9      Train_Loss: 0.0002      Train_Accuracy: 99.82      Test_Loss: 0.0006     Test_Accuracy: 99.50    Time_Duration: 42.8 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 10     Train_Loss: 0.0001      Train_Accuracy: 99.90      Test_Loss: 0.0007     Test_Accuracy: 99.44    Time_Duration: 42.9 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------------
ResNet Architecture Training and Testing Completed!
--------------------------------------------------
Model Training and Testing End date: 14/05/2021
Model Training and Testing End time: 23:23:14
--------------------------------------------------
```

Fig 13. ResNet Results on MNIST data

Note: Output plot of the final tensor values for MNIST data on ResNet



A.1.14 Individual accuracies of the image classes for MNIST in ResNet

```
--------------------------
Accuracy of     0 : 99 %
--------------------------
--------------------------
Accuracy of     1 : 98 %
--------------------------
--------------------------
Accuracy of     2 : 99 %
--------------------------
--------------------------
Accuracy of     3 : 98 %
--------------------------
--------------------------
Accuracy of     4 : 99 %
--------------------------
--------------------------
Accuracy of     5 : 98 %
--------------------------
--------------------------
Accuracy of     6 : 99 %
--------------------------
--------------------------
Accuracy of     7 : 100 %
--------------------------
--------------------------
Accuracy of     8 : 100 %
--------------------------
--------------------------
Accuracy of     9 : 100 %
--------------------------
```

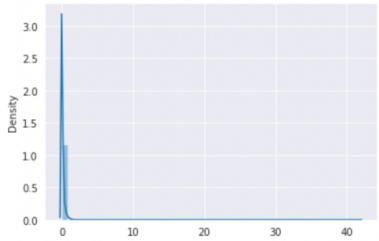
Fig 14. Individual accuracies of the image classes for MNIST in ResNet

A.1.15 GoogleNet Architecture Results on MNIST data

```
------------------------------------------------------
Model Training and Testing Start date: 14/05/2021
Model Training and Testing starting time: 23:15:52
------------------------------------------------------
GoogleNet Architecture Training and Testing Begins!
-----------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 1      Train_Loss: 0.0413      Train_Accuracy: 54.46      Test_Loss: 0.0039     Test_Accuracy: 96.48    Time_Duration: 43.5 sec
-----------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 2      Train_Loss: 0.0039      Train_Accuracy: 96.51      Test_Loss: 0.0023     Test_Accuracy: 97.79    Time_Duration: 43.4 sec
-----------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 3      Train_Loss: 0.0020      Train_Accuracy: 98.08      Test_Loss: 0.0013     Test_Accuracy: 98.73    Time_Duration: 43.4 sec
-----------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 4      Train_Loss: 0.0013      Train_Accuracy: 98.69      Test_Loss: 0.0015     Test_Accuracy: 98.58    Time_Duration: 43.5 sec
-----------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 5      Train_Loss: 0.0010      Train_Accuracy: 99.04      Test_Loss: 0.0017     Test_Accuracy: 98.43    Time_Duration: 43.5 sec
-----------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 6      Train_Loss: 0.0008      Train_Accuracy: 99.20      Test_Loss: 0.0012     Test_Accuracy: 98.81    Time_Duration: 43.4 sec
-----------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 7      Train_Loss: 0.0007      Train_Accuracy: 99.33      Test_Loss: 0.0011     Test_Accuracy: 98.96    Time_Duration: 43.5 sec
-----------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 8      Train_Loss: 0.0005      Train_Accuracy: 99.49      Test_Loss: 0.0012     Test_Accuracy: 99.00    Time_Duration: 43.4 sec
-----------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 9      Train_Loss: 0.0004      Train_Accuracy: 99.55      Test_Loss: 0.0017     Test_Accuracy: 98.41    Time_Duration: 43.5 sec
-----------------------------------------------------------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 10     Train_Loss: 0.0004      Train_Accuracy: 99.64      Test_Loss: 0.0014     Test_Accuracy: 98.81    Time_Duration: 43.4 sec
-----------------------------------------------------------------------------------------------------------------------------------------------
GoogleNet Architecture Training and Testing Completed!
------------------------------------------------------
Model Training and Testing End date: 14/05/2021
Model Training and Testing End time: 23:23:26
------------------------------------------------------
```

Fig 14. GoogleNet Results on MNIST data

Note: Output plot of final tensor values for MNIST data on GoogleNet



A.1.16 Individual accuracies of image classes for MNIST in GoogleNet

```
-----------------------------
Accuracy of     0 : 98 %
-----------------------------
-----------------------------
Accuracy of     1 : 100 %
-----------------------------
-----------------------------
Accuracy of     2 : 99 %
-----------------------------
-----------------------------
Accuracy of     3 : 98 %
-----------------------------
-----------------------------
Accuracy of     4 : 99 %
-----------------------------
-----------------------------
Accuracy of     5 : 99 %
-----------------------------
-----------------------------
Accuracy of     6 : 97 %
-----------------------------
-----------------------------
Accuracy of     7 : 100 %
-----------------------------
-----------------------------
Accuracy of     8 : 98 %
-----------------------------
-----------------------------
Accuracy of     9 : 100 %
-----------------------------
```

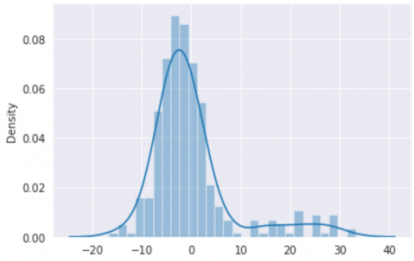
Fig 16. Individual accuracies of the image classes for MNIST in GoogleNet

A.1.17 AlexNet Architecture Results on MNIST data

```
Model Training and Testing Start date: 14/05/2021
Model Training and Testing starting time: 23:25:24
AlexNet Architecture Training and Testing Begins!
Epoch--> 1      Train_Loss: 0.0099      Train_Accuracy: 89.28      Test_Loss: 0.0015     Test_Accuracy: 98.73    Time_Duration: 40.5 sec
Epoch--> 2      Train_Loss: 0.0015      Train_Accuracy: 98.59      Test_Loss: 0.0008     Test_Accuracy: 99.26    Time_Duration: 40.0 sec
Epoch--> 3      Train_Loss: 0.0010      Train_Accuracy: 99.03      Test_Loss: 0.0007     Test_Accuracy: 99.26    Time_Duration: 39.8 sec
Epoch--> 4      Train_Loss: 0.0007      Train_Accuracy: 99.27      Test_Loss: 0.0006     Test_Accuracy: 99.35    Time_Duration: 39.4 sec
Epoch--> 5      Train_Loss: 0.0006      Train_Accuracy: 99.41      Test_Loss: 0.0008     Test_Accuracy: 99.19    Time_Duration: 39.8 sec
Epoch--> 6      Train_Loss: 0.0005      Train_Accuracy: 99.52      Test_Loss: 0.0007     Test_Accuracy: 99.32    Time_Duration: 39.5 sec
Epoch--> 7      Train_Loss: 0.0004      Train_Accuracy: 99.63      Test_Loss: 0.0007     Test_Accuracy: 99.38    Time_Duration: 39.6 sec
Epoch--> 8      Train_Loss: 0.0004      Train_Accuracy: 99.64      Test_Loss: 0.0007     Test_Accuracy: 99.40    Time_Duration: 39.8 sec
Epoch--> 9      Train_Loss: 0.0003      Train_Accuracy: 99.68      Test_Loss: 0.0007     Test_Accuracy: 99.32    Time_Duration: 40.4 sec
Epoch--> 10     Train_Loss: 0.0002      Train_Accuracy: 99.78      Test_Loss: 0.0006     Test_Accuracy: 99.40    Time_Duration: 39.6 sec
AlexNet Architecture Training and Testing Completed!
Model Training and Testing End date: 14/05/2021
Model Training and Testing End time: 23:32:19
```

Fig 17. AlexNet Results on MNIST data

Note: Output plot of the final tensor values for MNIST data on AlexNet



A.1.18 Individual accuracies of image classes for MNIST in AlexNet

```
Accuracy of     0 : 99 %
Accuracy of     1 : 98 %
Accuracy of     2 : 100 %
Accuracy of     3 : 98 %
Accuracy of     4 : 99 %
Accuracy of     5 : 98 %
Accuracy of     6 : 100 %
Accuracy of     7 : 99 %
Accuracy of     8 : 100 %
Accuracy of     9 : 99 %
```

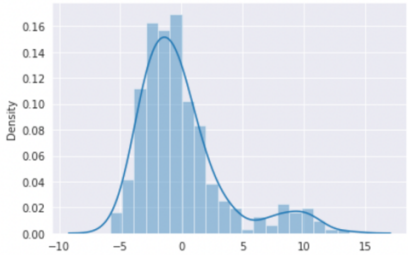
Fig 18. Individual accuracies of the image classes for MNIST in AlexNet

A.1.19 MobileNet Architecture Results on MNIST data

```
Model Training and Testing Start date: 14/05/2021
Model Training and Testing starting time: 23:35:05
MobileNet Architecture Training and Testing Begins!
Epoch--> 1      Train_Loss: 0.0119      Train_Accuracy: 89.70      Test_Loss: 0.0034     Test_Accuracy: 97.77    Time_Duration: 101.1 sec
Epoch--> 2      Train_Loss: 0.0035      Train_Accuracy: 97.14      Test_Loss: 0.0019     Test_Accuracy: 98.65    Time_Duration: 101.1 sec
Epoch--> 3      Train_Loss: 0.0022      Train_Accuracy: 98.20      Test_Loss: 0.0019     Test_Accuracy: 98.80    Time_Duration: 101.0 sec
Epoch--> 4      Train_Loss: 0.0017      Train_Accuracy: 98.56      Test_Loss: 0.0023     Test_Accuracy: 98.67    Time_Duration: 101.0 sec
Epoch--> 5      Train_Loss: 0.0013      Train_Accuracy: 98.90      Test_Loss: 0.0029     Test_Accuracy: 97.74    Time_Duration: 101.6 sec
Epoch--> 6      Train_Loss: 0.0011      Train_Accuracy: 99.03      Test_Loss: 0.0013     Test_Accuracy: 99.05    Time_Duration: 101.7 sec
Epoch--> 7      Train_Loss: 0.0009      Train_Accuracy: 99.21      Test_Loss: 0.0012     Test_Accuracy: 99.02    Time_Duration: 101.7 sec
Epoch--> 8      Train_Loss: 0.0009      Train_Accuracy: 99.19      Test_Loss: 0.0011     Test_Accuracy: 99.22    Time_Duration: 101.7 sec
Epoch--> 9      Train_Loss: 0.0008      Train_Accuracy: 99.31      Test_Loss: 0.0012     Test_Accuracy: 99.00    Time_Duration: 101.7 sec
Epoch--> 10     Train_Loss: 0.0006      Train_Accuracy: 99.49      Test_Loss: 0.0012     Test_Accuracy: 99.18    Time_Duration: 101.6 sec
MobileNet Architecture Training and Testing Completed!
Model Training and Testing End date: 14/05/2021
Model Training and Testing End time: 23:52:20
```

Fig 19. MobileNet Results on MNIST data

Note: Output plot of final tensor values for MNIST data on MobileNet



A.1.20 Individual accuracies of image classes for MNIST in MobileNet

```
Accuracy of     0 : 97 %
Accuracy of     1 : 100 %
Accuracy of     2 : 100 %
Accuracy of     3 : 98 %
Accuracy of     4 : 99 %
Accuracy of     5 : 98 %
Accuracy of     6 : 99 %
Accuracy of     7 : 99 %
Accuracy of     8 : 99 %
Accuracy of     9 : 98 %
```

Fig 20. Individual accuracies of the image classes for MNIST in MobileNet

Note: Code for Initialisation of weights

```
#Initialize the Weights and display the Network architecture
def Initialise_weights(layer_weight):
    if type(layer_weight) == torch.nn.Conv2d:
        torch.nn.init.xavier_uniform_(layer_weight.weight)
Network_arch.apply(Initialise_weights)
```

A.1.21 VGG16 Results on CIFAR10 data

```
Epoch--> 28     Train_Loss: 0.0021     Train_Accuracy: 90.41     Test_Loss: 0.0044     Test_Accuracy: 82.94     Time_Duration: 24.2 sec
Epoch--> 29     Train_Loss: 0.0020     Train_Accuracy: 90.94     Test_Loss: 0.0036     Test_Accuracy: 85.45     Time_Duration: 24.2 sec
Epoch--> 30     Train_Loss: 0.0019     Train_Accuracy: 91.46     Test_Loss: 0.0036     Test_Accuracy: 85.81     Time_Duration: 24.1 sec
Epoch--> 31     Train_Loss: 0.0018     Train_Accuracy: 91.97     Test_Loss: 0.0040     Test_Accuracy: 84.78     Time_Duration: 24.1 sec
Epoch--> 32     Train_Loss: 0.0017     Train_Accuracy: 92.22     Test_Loss: 0.0038     Test_Accuracy: 85.21     Time_Duration: 24.1 sec
Epoch--> 33     Train_Loss: 0.0017     Train_Accuracy: 92.18     Test_Loss: 0.0048     Test_Accuracy: 83.00     Time_Duration: 24.1 sec
Epoch--> 34     Train_Loss: 0.0016     Train_Accuracy: 92.85     Test_Loss: 0.0045     Test_Accuracy: 84.43     Time_Duration: 24.1 sec
Epoch--> 35     Train_Loss: 0.0016     Train_Accuracy: 92.98     Test_Loss: 0.0040     Test_Accuracy: 85.36     Time_Duration: 24.1 sec
Epoch--> 36     Train_Loss: 0.0015     Train_Accuracy: 93.22     Test_Loss: 0.0035     Test_Accuracy: 86.27     Time_Duration: 24.1 sec
Epoch--> 37     Train_Loss: 0.0015     Train_Accuracy: 93.43     Test_Loss: 0.0045     Test_Accuracy: 83.46     Time_Duration: 24.0 sec
Epoch--> 38     Train_Loss: 0.0014     Train_Accuracy: 93.53     Test_Loss: 0.0040     Test_Accuracy: 85.20     Time_Duration: 24.1 sec
Epoch--> 39     Train_Loss: 0.0013     Train_Accuracy: 94.14     Test_Loss: 0.0036     Test_Accuracy: 86.15     Time_Duration: 23.7 sec
Epoch--> 40     Train_Loss: 0.0013     Train_Accuracy: 94.18     Test_Loss: 0.0041     Test_Accuracy: 85.52     Time_Duration: 24.2 sec
VGG16 Architecture Training and Testing Completed!
Model Training and Testing End date: 14/05/2021
Model Training and Testing End time: 23:59:24
```

Fig 21. VGG16 Results on CIFAR10 data

A.1.22 Original and Predicted image classes for CIFAR10 on VGG16

GROUNDTRUTH IMAGE

```
Orginal images classes:  horse dog ship plane ship bird
Predicted images classes:  horse dog ship plane ship deer
```

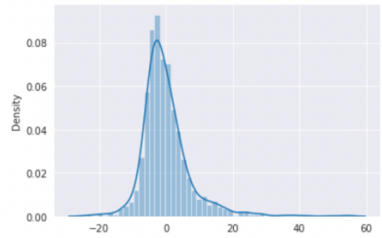
Fig 22. Original and Predicted image classes for CIFAR10 on VGG16

Note: Wrong Predictions on CIFAR10 data(Failure)

GROUNDTRUTH IMAGE

```
Orginal images classes:  horse dog ship plane ship bird
Predicted images classes:  deer frog plane plane car deer
```

Note: Output plot of the final tensor values for CIFAR10 data on VGG16



A.1.23 Individual accuracies of image classes for CIFAR10 on VGG16

```
--------------------------------
Accuracy of plane : 95 %
--------------------------------
--------------------------------
Accuracy of   car : 93 %
--------------------------------
--------------------------------
Accuracy of  bird : 79 %
--------------------------------
--------------------------------
Accuracy of   cat : 77 %
--------------------------------
--------------------------------
Accuracy of  deer : 85 %
--------------------------------
--------------------------------
Accuracy of   dog : 81 %
--------------------------------
--------------------------------
Accuracy of  frog : 91 %
--------------------------------
--------------------------------
Accuracy of horse : 87 %
--------------------------------
--------------------------------
Accuracy of  ship : 91 %
--------------------------------
--------------------------------
Accuracy of truck : 92 %
--------------------------------
```

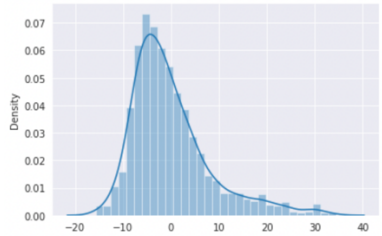
Fig 23. Individual accuracies of the image classes for CIFAR10 in VGG16

A.1.24 ResNet Results on CIFAR10 data

```
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 28     Train_Loss: 0.0009     Train_Accuracy: 96.12     Test_Loss: 0.0042     Test_Accuracy: 86.23     Time_Duration: 59.5 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 29     Train_Loss: 0.0008     Train_Accuracy: 96.33     Test_Loss: 0.0034     Test_Accuracy: 88.71     Time_Duration: 59.5 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 30     Train_Loss: 0.0008     Train_Accuracy: 96.43     Test_Loss: 0.0032     Test_Accuracy: 89.55     Time_Duration: 59.4 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 31     Train_Loss: 0.0007     Train_Accuracy: 96.65     Test_Loss: 0.0034     Test_Accuracy: 88.98     Time_Duration: 59.5 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 32     Train_Loss: 0.0007     Train_Accuracy: 96.97     Test_Loss: 0.0053     Test_Accuracy: 85.40     Time_Duration: 59.5 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 33     Train_Loss: 0.0006     Train_Accuracy: 97.16     Test_Loss: 0.0034     Test_Accuracy: 89.66     Time_Duration: 59.4 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 34     Train_Loss: 0.0006     Train_Accuracy: 97.13     Test_Loss: 0.0038     Test_Accuracy: 88.13     Time_Duration: 59.4 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 35     Train_Loss: 0.0006     Train_Accuracy: 97.29     Test_Loss: 0.0037     Test_Accuracy: 89.12     Time_Duration: 59.4 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 36     Train_Loss: 0.0006     Train_Accuracy: 97.51     Test_Loss: 0.0069     Test_Accuracy: 83.18     Time_Duration: 59.4 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 37     Train_Loss: 0.0006     Train_Accuracy: 97.47     Test_Loss: 0.0034     Test_Accuracy: 89.95     Time_Duration: 59.4 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 38     Train_Loss: 0.0005     Train_Accuracy: 97.73     Test_Loss: 0.0036     Test_Accuracy: 89.60     Time_Duration: 59.4 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 39     Train_Loss: 0.0005     Train_Accuracy: 97.78     Test_Loss: 0.0036     Test_Accuracy: 89.49     Time_Duration: 59.4 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 40     Train_Loss: 0.0005     Train_Accuracy: 97.91     Test_Loss: 0.0040     Test_Accuracy: 88.84     Time_Duration: 59.4 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
ResNet Architecture Training and Testing Completed!
--------------------------------------------------
Model Training and Testing End date: 15/05/2021
Model Training and Testing End time: 00:43:03
--------------------------------------------------
```

Fig 24. ResNet Results on CIFAR10 data

Note: Output plot of final tensor values for CIFAR10 data on ResNet



A.1.25 Individual accuracies of image classes for CIFAR10 on ResNet

```
--------------------------------
Accuracy of plane : 90 %
--------------------------------
--------------------------------
Accuracy of   car : 97 %
--------------------------------
--------------------------------
Accuracy of  bird : 79 %
--------------------------------
--------------------------------
Accuracy of   cat : 88 %
--------------------------------
--------------------------------
Accuracy of  deer : 95 %
--------------------------------
--------------------------------
Accuracy of   dog : 71 %
--------------------------------
--------------------------------
Accuracy of  frog : 85 %
--------------------------------
--------------------------------
Accuracy of horse : 89 %
--------------------------------
--------------------------------
Accuracy of  ship : 98 %
--------------------------------
--------------------------------
Accuracy of truck : 92 %
--------------------------------
```

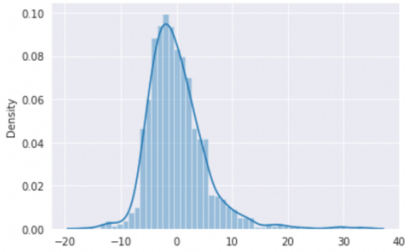
Fig 25. Individual accuracies of the image classes for CIFAR10 in ResNetNet

A.1.26 GoogleNet Results on CIFAR10 data

```
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 28     Train_Loss: 0.0046     Train_Accuracy: 79.54     Test_Loss: 0.0052     Test_Accuracy: 78.29     Time_Duration: 41.3 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 29     Train_Loss: 0.0044     Train_Accuracy: 80.25     Test_Loss: 0.0077     Test_Accuracy: 70.24     Time_Duration: 41.3 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 30     Train_Loss: 0.0042     Train_Accuracy: 81.33     Test_Loss: 0.0063     Test_Accuracy: 74.82     Time_Duration: 41.4 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 31     Train_Loss: 0.0041     Train_Accuracy: 81.97     Test_Loss: 0.0059     Test_Accuracy: 75.33     Time_Duration: 41.3 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 32     Train_Loss: 0.0039     Train_Accuracy: 82.63     Test_Loss: 0.0055     Test_Accuracy: 77.46     Time_Duration: 41.3 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 33     Train_Loss: 0.0038     Train_Accuracy: 83.41     Test_Loss: 0.0049     Test_Accuracy: 79.23     Time_Duration: 41.3 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 34     Train_Loss: 0.0036     Train_Accuracy: 83.81     Test_Loss: 0.0058     Test_Accuracy: 76.38     Time_Duration: 41.3 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 35     Train_Loss: 0.0034     Train_Accuracy: 84.66     Test_Loss: 0.0047     Test_Accuracy: 80.54     Time_Duration: 41.2 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 36     Train_Loss: 0.0034     Train_Accuracy: 85.03     Test_Loss: 0.0055     Test_Accuracy: 77.80     Time_Duration: 41.3 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 37     Train_Loss: 0.0032     Train_Accuracy: 86.06     Test_Loss: 0.0053     Test_Accuracy: 78.80     Time_Duration: 41.3 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 38     Train_Loss: 0.0031     Train_Accuracy: 86.29     Test_Loss: 0.0069     Test_Accuracy: 74.61     Time_Duration: 41.3 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 39     Train_Loss: 0.0030     Train_Accuracy: 86.88     Test_Loss: 0.0062     Test_Accuracy: 77.75     Time_Duration: 41.4 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------------------------------------------------------------------
Epoch--> 40     Train_Loss: 0.0029     Train_Accuracy: 87.29     Test_Loss: 0.0058     Test_Accuracy: 77.00     Time_Duration: 41.4 sec
--------------------------------------------------------------------------------------------------------------------------------------------------------
GoogleNet Architecture Training and Testing Completed!
--------------------------------------------------
Model Training and Testing End date: 15/05/2021
Model Training and Testing End time: 00:31:02
--------------------------------------------------
```

Fig 26. GoogleNet Results on CIFAR10 data

Note: Output plot of final tensor values for CIFAR10 data on GoogleNet



A.1.27 Individual accuracies of image class for CIFAR10 on GoogleNet

```
--------------------------------------------------
Accuracy of plane : 76 %
--------------------------------------------------
--------------------------------------------------
Accuracy of   car : 87 %
--------------------------------------------------
--------------------------------------------------
Accuracy of  bird : 66 %
--------------------------------------------------
--------------------------------------------------
Accuracy of   cat : 90 %
--------------------------------------------------
--------------------------------------------------
Accuracy of  deer : 74 %
--------------------------------------------------
--------------------------------------------------
Accuracy of   dog : 53 %
--------------------------------------------------
--------------------------------------------------
Accuracy of  frog : 68 %
--------------------------------------------------
--------------------------------------------------
Accuracy of horse : 83 %
--------------------------------------------------
--------------------------------------------------
Accuracy of  ship : 89 %
--------------------------------------------------
--------------------------------------------------
Accuracy of truck : 96 %
--------------------------------------------------
```

Fig 27. Individual accuracies of the image classes for CIFAR10 in GoogleNet

A.1.28 Comparison table between all the developed Deep Network Models Including AlexNet and MobileNet

Model	Dataset Used	No of Epochs	Train Loss	Final Train Accuracy	Test Loss	Test Accuracy	Training Time(Approx)
VGG16	MNIST	10	0.0001	99.86	0.0007	99.25	43 Minutes
ResNet	MNIST	10	0.0001	99.90	0.0007	99.44	7 Minutes
GoogleNet	MNIST	10	0.0004	99.64	0.0014	98.81	7 Minutes
AlexNet	MNIST	10	0.0002	99.78	0.0006	99.40	7 Minutes
MobileNet	MNIST	10	0.0006	99.49	0.0012	99.18	17 Minutes
VGG16	CIFAR10	40	0.0013	94.18	0.0041	85.52	16 Minutes
ResNet	CIFAR10	40	0.0005	99.91	0.0040	88.84	40 Minutes
GoogleNet	CIFAR10	40	0.0029	87.29	0.0058	77.00	28 Minutes

Fig 28. Comparison table between the developed deep network models which includes the outputs of AlexNet and MobileNet

In MNIST data, all the developed models which includes VGG16, ResNet, GoogleNet, AlexNet and MobileNet reaches 99% test accuracy, which leads to accurate prediction of data in that specific dataset. Whereas when the model is futher evaluated in the CIFAR10 dataset, VGG and the GoogleNet models achieves 80% (approx) test accuracy but the ResNet model achieves 90% accuracy which is higher the both the above specified model. In CIFAR10 dataset, the ResNet model work much better in means of achieving better accuracy which also leads to the drop in the loss to the larger extent than both VGG and the GoogleNet models. From this, it is concluded that the ResNet model developed works better than other developed models.

A.1.29 Differential of Sigmoid Function

$$f(x) = \frac{1}{1+e^{-x}}$$

$$f'(x) = \frac{e^{-x}}{(1+e^{-x})^2}$$

$$= \frac{1}{1+e^{-x}} \cdot \frac{e^{-x}}{1+e^{-x}}$$

$$= \frac{1}{1+e^{-x}} \left(1 - \frac{1}{1+e^{-x}}\right)$$

$$f'(x) = f(x)(1-f(x))$$

A.1.30 Differential of Softmax Function

$$f_i = f(i; c_1, c_2 \cdots c_o)$$

$$= \frac{e^{c_i}}{\sum_i e^{c_i}}$$

To find the differential of softmax function, there are 2 cases.

1) $\frac{df_i}{dc_k}$, where $i = k$

2) $\frac{df_i}{dc_k}$, where $i \neq k$

$$f_1 = f(1; c_1, c_2) = \frac{e^{c_1}}{e^{c_1} + e^{c_2}}$$

case ① $(i = k)$

$$\frac{df_1}{dc_1} = \frac{(e^{c_1}+e^{c_2})e^{c_1} - e^{c_1}e^{c_1}}{(e^{c_1}+e^{c_2})^2}$$

$$= \frac{e^{c_1}}{(e^{c_1}+e^{c_2})} - \left(\frac{e^{c_1}}{(e^{c_1}+e^{c_2})}\right)^2$$

$$= \frac{e^{c_1}}{(e^{c_1}+e^{c_2})}\left(1 - \frac{e^{c_1}}{(e^{c_1}+e^{c_2})}\right)$$

$$\frac{df_1}{dc_1} = f_1(1-f_1) \rightarrow ①$$

case ② $(i \neq k)$

$$\frac{df_1}{dc_2} = -\frac{e^{c_1}e^{c_2}}{(e^{c_1}+e^{c_2})^2}$$

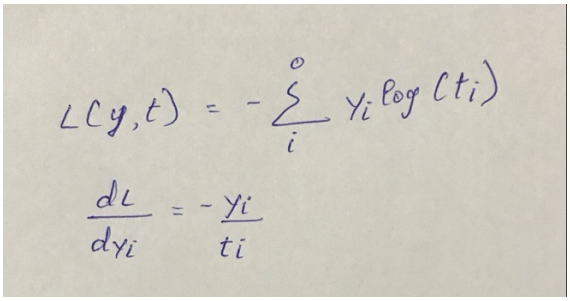
$$\frac{df_1}{dc_2} = -f_1 f_2 \rightarrow ②$$

Generalizing, it gives;

$$\frac{df_i}{dc_i} = \begin{cases} f_i(1-f_i) & i = k \\ -f_i f_k & i \neq k \end{cases} \rightarrow ③$$

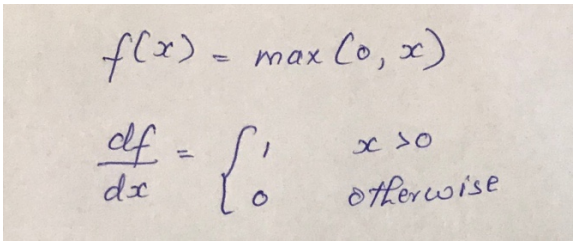
A.1.31 Differential of Cross Entropy Loss

The Loss between the output vector and the target vector is given by

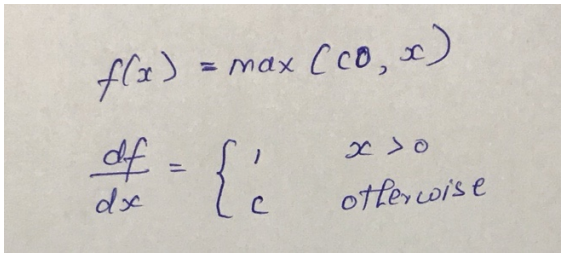


A.1.32 Differential of Rectified Linear Unit (ReLU)

ReLU function is defined as

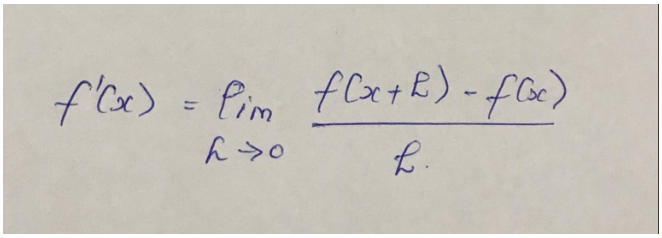


Leaky ReLU is defined as



A.1.33 Differential of Max-Pooling

The differential of max pooling is given by



A.1.34 Mathematical derivation of Back-Propagation

A.1.34.1. Backpropagation from Hidden layer to Output Layer

Input =X, Target =t, Output =c

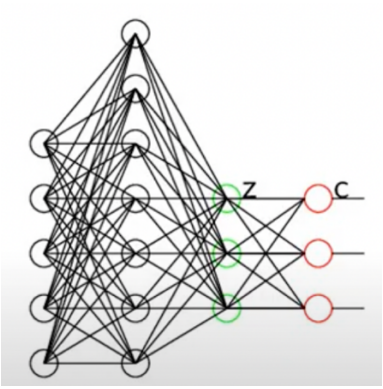


Fig 29. Backpropagated neural Nets

DERIVATION:

$$\frac{dL}{dv_{jk}} = \frac{dL}{dz_k} \frac{dz_k}{dv_{jk}}$$

$$\frac{dL}{dz_k} = \sum_{o=1}^{o} \frac{dL}{dc_o} \frac{dc_o}{dz_k}$$

$$= \frac{dL}{dc_k} \frac{dc_k}{dz_k} + \sum_{o \neq k} \frac{dL}{dc_o} \frac{dc_o}{dz_k}$$

$$= \frac{-t_k}{c_k} c_k (1-c_k) + \sum_{o \neq k} \frac{-t_o}{c_o} (-c_o c_k)$$

$$= -t_k (1-c_k) + \sum_{o \neq k} t_o c_k$$

$$= -t_k + t_k c_k + \sum_{o \neq k} t_o c_k$$

$$= -t_k + \sum_{o \neq k} t_o c_k + t_k c_k$$

$$= -t_k + c_k \sum_{o} t_o$$

$$= -t_k + c_k 1$$

$$\frac{dL}{dz_k} = c_k - t_k$$

$$\frac{dz_k}{dv_{jk}} = b_j$$

$$dv_{jk} = (c_k - t_k) b_j$$

A.1.34.2. Backpropagation from Input layer to Hidden Layer

$$\frac{dL}{du_{ij}} = \sum_{k=0}^{k} \frac{dL}{dz_k} \frac{dz_k}{db_j} \frac{db_j}{dy_j} \frac{dy_j}{du_{ij}}$$

$$= \frac{db_j}{dy_j} \frac{dy_j}{du_{ij}} \sum_{k} \frac{dL}{dz_k} \frac{dz_k}{db_j}$$

$$= b_j (1 - b_j) x_i \sum_{k} (c_k - t_k) v_{jk}$$

A.1.34.3. Backpropagation from Kernels

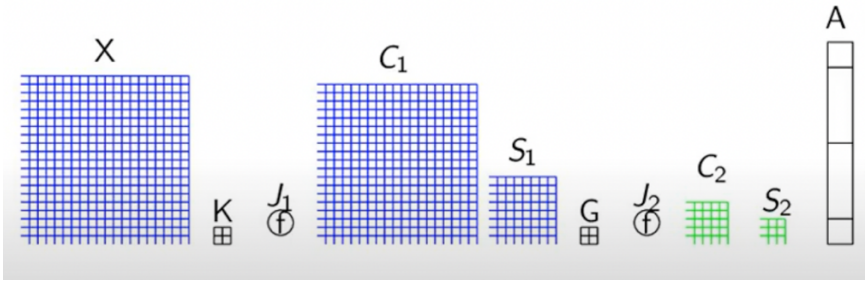
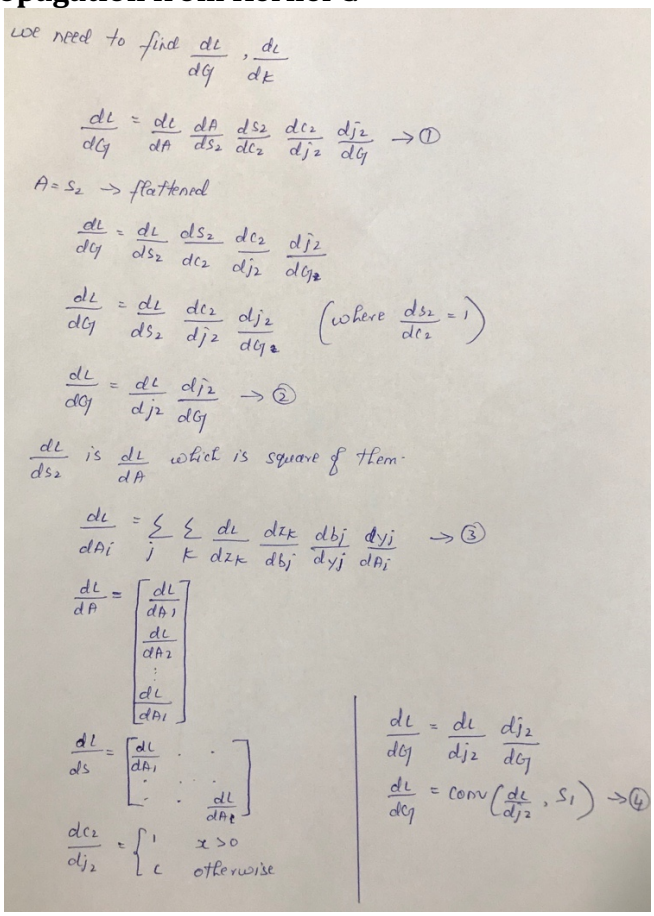
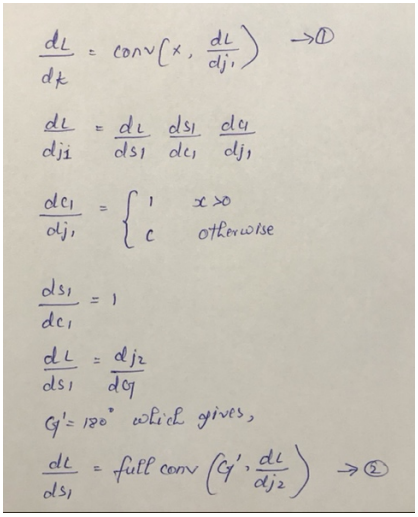


Fig 30. Backpropagated through kernels

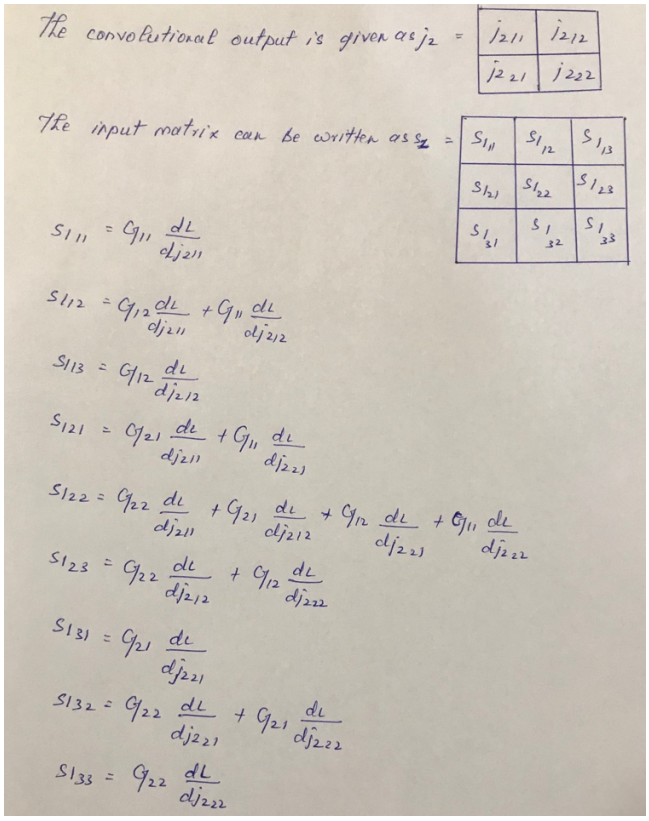
A.1.34.3.1 Backpropagation from Kernel G



A.1.34.3.2 Backpropagation from Kernel K



A.1.34.4. Full Convolution



Note: Google Colaboratory-Pro GPU Usage

```
Sat May 15 00:14:03 2021
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 465.19.01    Driver Version: 460.32.03    CUDA Version: 11.2     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  Tesla P100-PCIE...  Off  | 00000000:00:04.0 Off |                    0 |
| N/A   35C    P0    34W / 250W |   6219MiB / 16280MiB |      0%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
+-----------------------------------------------------------------------------+
```

Appendix – A1: Outputs from the Tensorboard (Done for Extra Visualisation)

I have also Created a Two supplementary Python files, in which I have trained models and stored the outputs in my Google drive using tensorboard. MNIST in One file and the CIFAR10 In the Other File. In these, two files I have used Tensorboard to store the plot and the predicted outputs. In this I have loaded the models from Pytorch models library and fixed **Pretrained=False**.

Supplementary Extra Notebook Names in which tensorboard is used

File uses MNIST dataset

- MiniProject_MNIST-models_Tensorboard_200434339_Ganesh_Kumaran_Masilamani.py

File uses CIFAR10 dataset

- MiniProject_CIFAR10-models_Tensorboard_200434339_Ganesh_Kumaran_Masilamani.py

A1.1 Initialization of the Tensorboard

```
#Initialisation of the tensorboard
%load_ext tensorboard
%tensorboard --logdir drive/MyDrive/Outputs/Mini_project/outs/
```

Fig A1.1. Code to Initialise tensorboard

A1.2 TensorBoard display

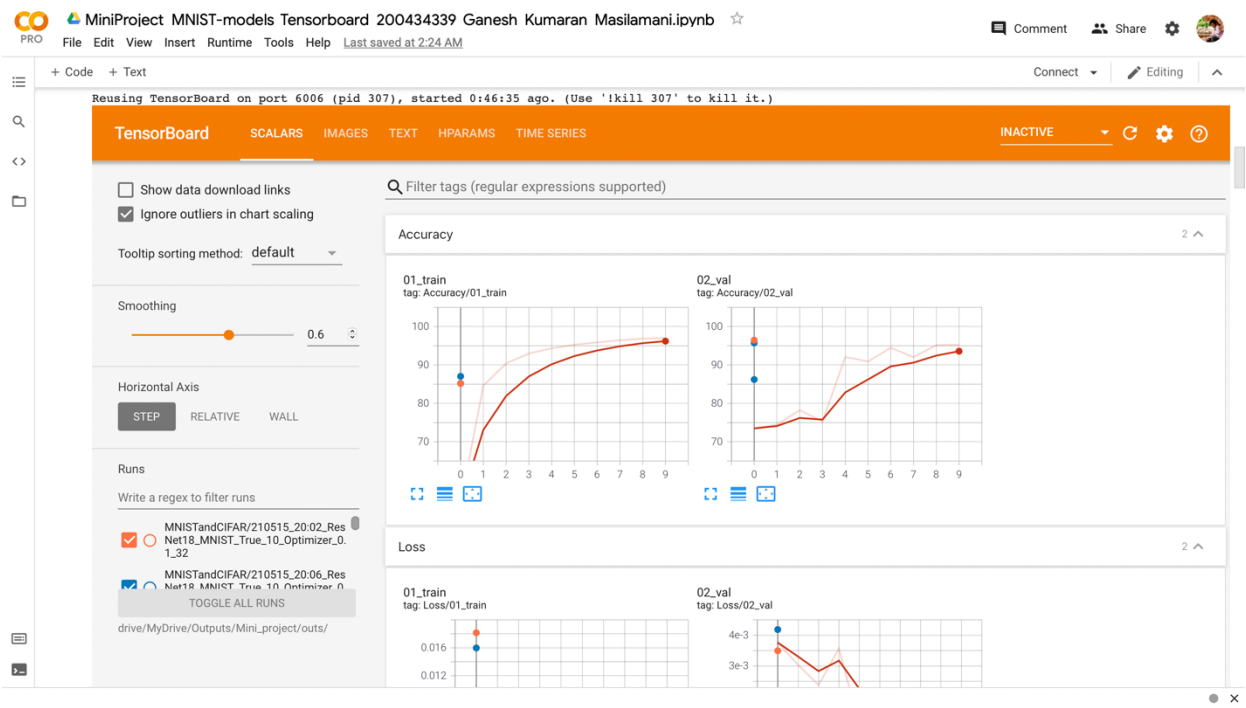


Fig A1.2. Display of the Tensorboard

A1.3 Example of MNIST Correct Prediction Stored in tensorboard

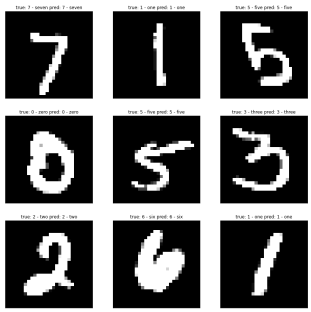


Fig A1.3. MNIST correct prediction

A1.4 Example of MNIST Incorrect Prediction Stored in tensorboard

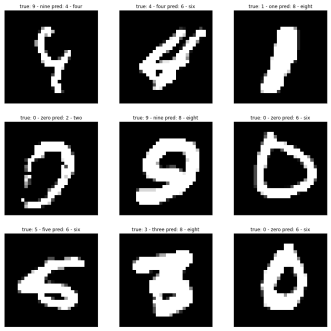


Fig A1.4. MNIST Incorrect prediction

A1.5 Example of Heatmap of the Confusion Matrix Stored in tensorboard

This plot is the Confusion matrix formed of the final tensor value which is obtained after training and evaluation.

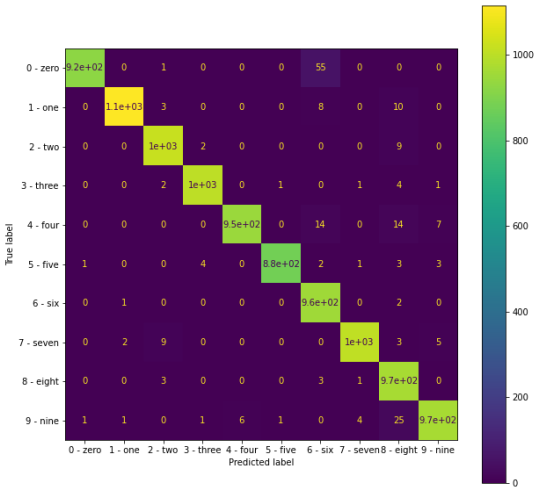


Fig A1.5. Confusion matrix when using MNIST dataset

A1.6 Example of CIFAR10 Correct Prediction Stored in tensorboard

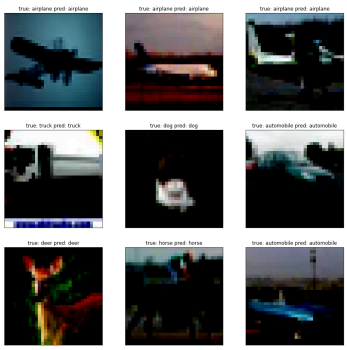


Fig A1.6. CIFAR10 correct prediction

A1.7 Example of CIFAR10 Incorrect Prediction Stored in tensorboard

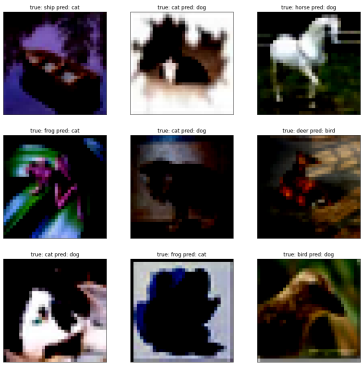


Fig A1.7. CIFAR10 Incorrect prediction

A1.8 Example of Heatmap of the Confusion Matrix Stored in tensorboard

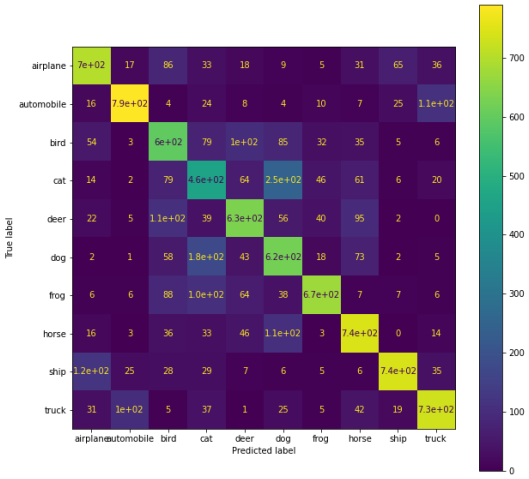


Fig A1.8. Confusion matrix when using CIFAR10 dataset

A1.9 Example of Train and Validation accuracy plot Stored in tensorboard

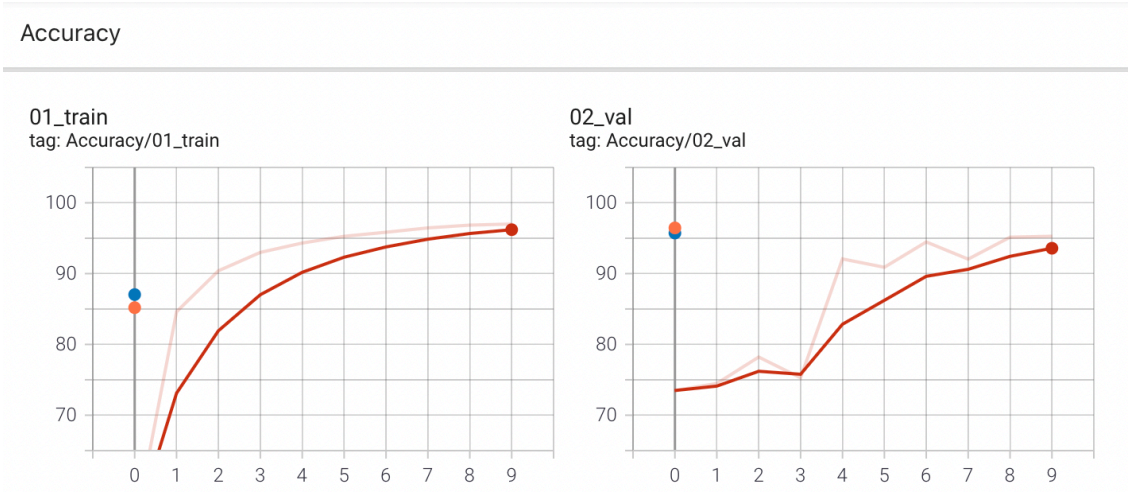


Fig A1.9. plot for Train and Validation accuracy

A1.10 Example of Train and Validation Loss plot Stored in tensorboard

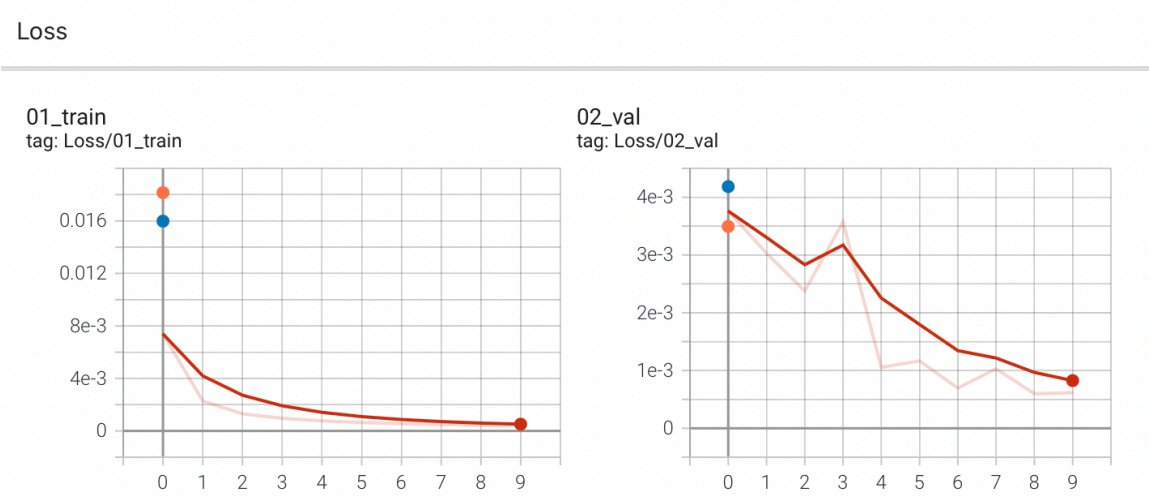


Fig A1.10. plot for Train and Validation Loss