Introduction

Here I have designed a network that combines supervised and unsupervised architectures in one model to achieve a classification on CIFAR-10 datasets. So, for this I build a encoder model and this model compressed the image data and after decoder model decompressed the data and again this model reconstruct the original image.

Another part of this notebook is to used the Pre-Training CNNs Using Convolutional Autoencoders to classify the image.

For completing this project/ assignment I have followed some module those are:

- `Tensorflow`
- `Keras`
- `Scikit-learn`
- `Pandas`
- `Numpy`

The CIFAR10 dataset contains 60,000, 32×32 pixel color images in 10 classes, with
6,000 images in each class. The dataset is divided into 50,000 training images and 10,000 testing images. The classes are mutually exclusive and there is no overlap between them.

The class labels and their standard associated integer values are listed below.

- `0 --> **airplane**`
- `1 --> **automobile**`
- `2 --> **bird**`
- `3 --> **cat**`
- `4 --> **deer**`
- `5 --> **dog**`
- `6 --> **frog**`
- `7 --> **horse**`
- `8 --> **ship**`
- `9 --> **truck**`

Spliting Datasets into four parts 👀

- `1. X_train`
- `2. X_test`
- `3. Y_train`
- `4. Y_test`

```
 ------------------------------
 It is clear that the images of the datasets are indeed very small
compared to modern photographs; it can be challenging to see what
exactly is represented in some of the images given the extremely low
```

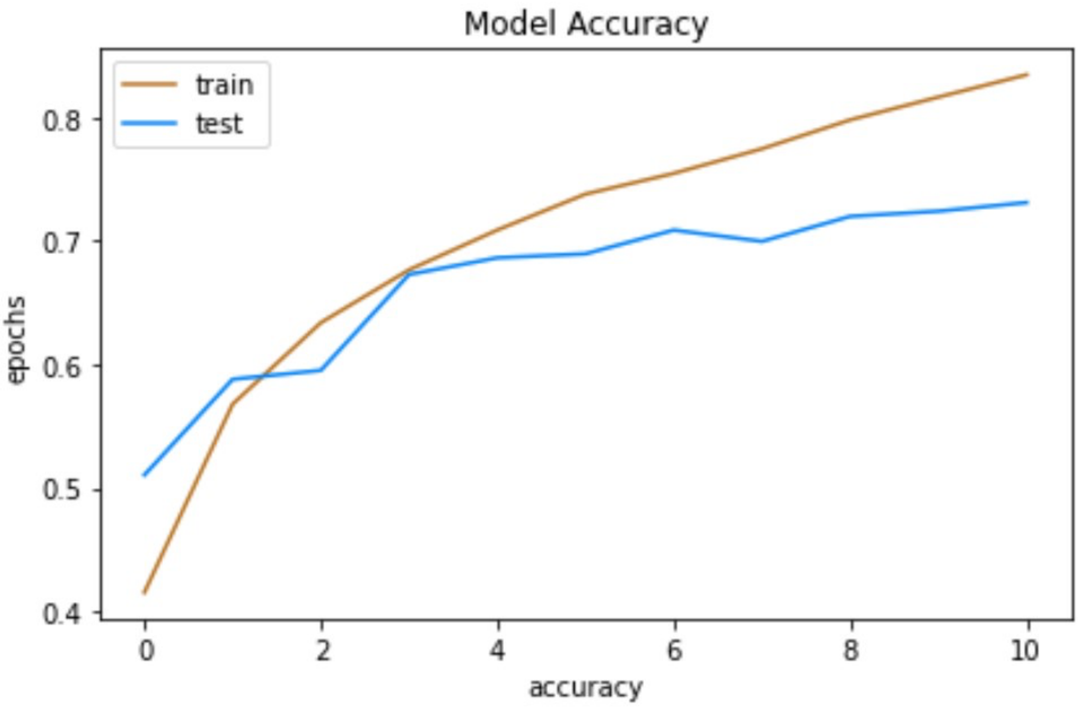
`resolution.This low resolution is likely the cause of the limited performance that top-of-the-line algorithms are able to achieve on the dataset.`

The max pixel value is 255 for each channel. Normalize the images to a number from 0 to 1. Image has 3 channels (R,G,B) and each value in the channel can range from 0 to 255.
Hence to normalize in 0-->1 range, we need to divide it by 255

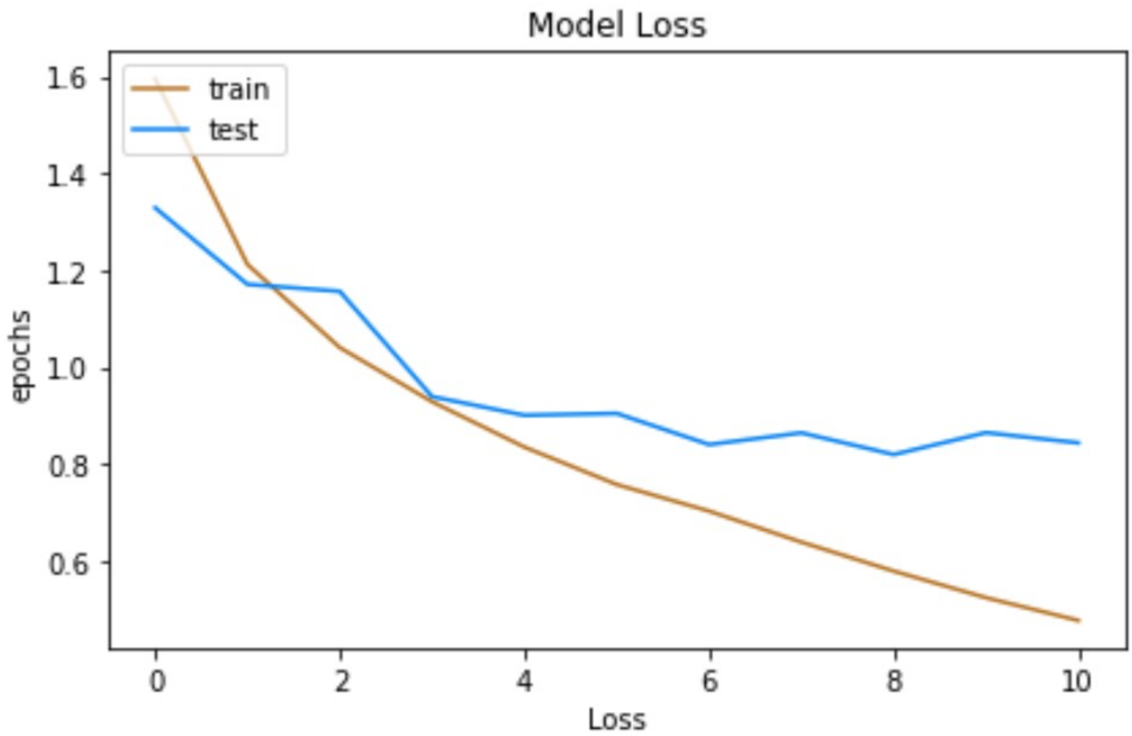
I have used 4 different optimizers into my task for compare the accuracy and loss values

- `1.Adam`
- `2.Nadam`
- `3.Rmsprop`
- `4.SGD`

Model accuracy with **`Adam`** Optimizers:

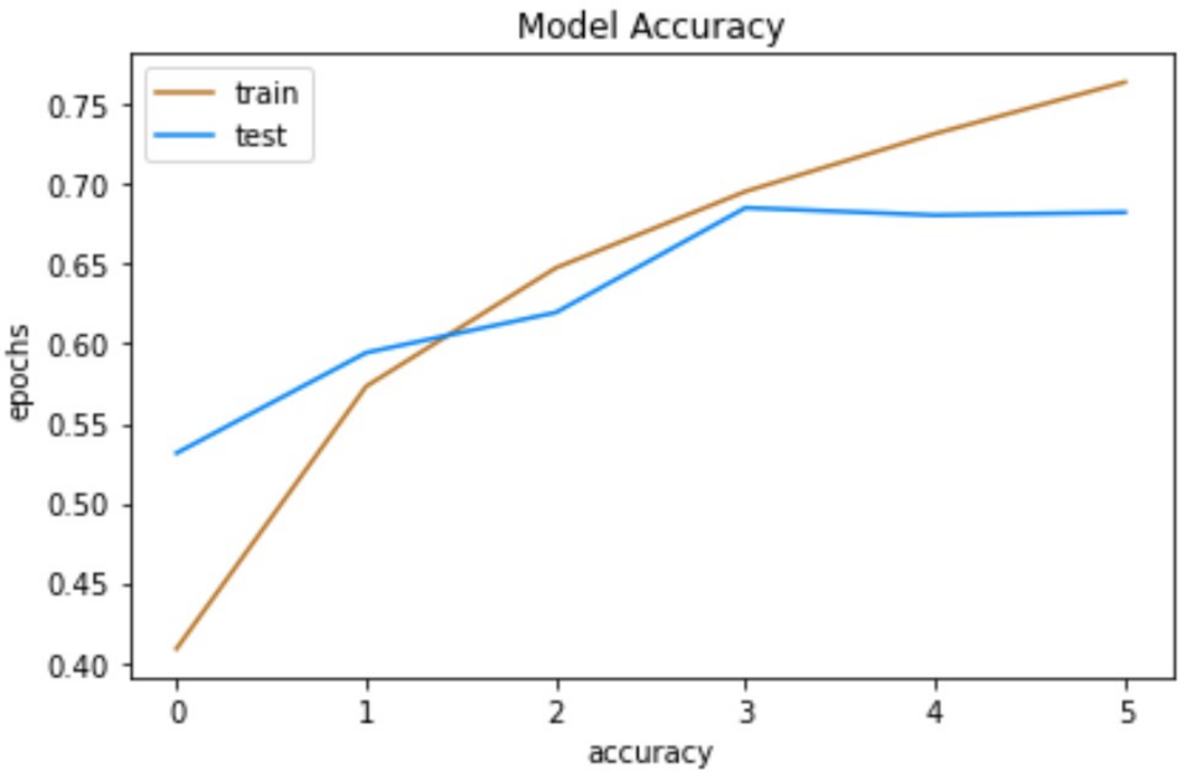


Model Loss with **`Adam`** Optimizers:



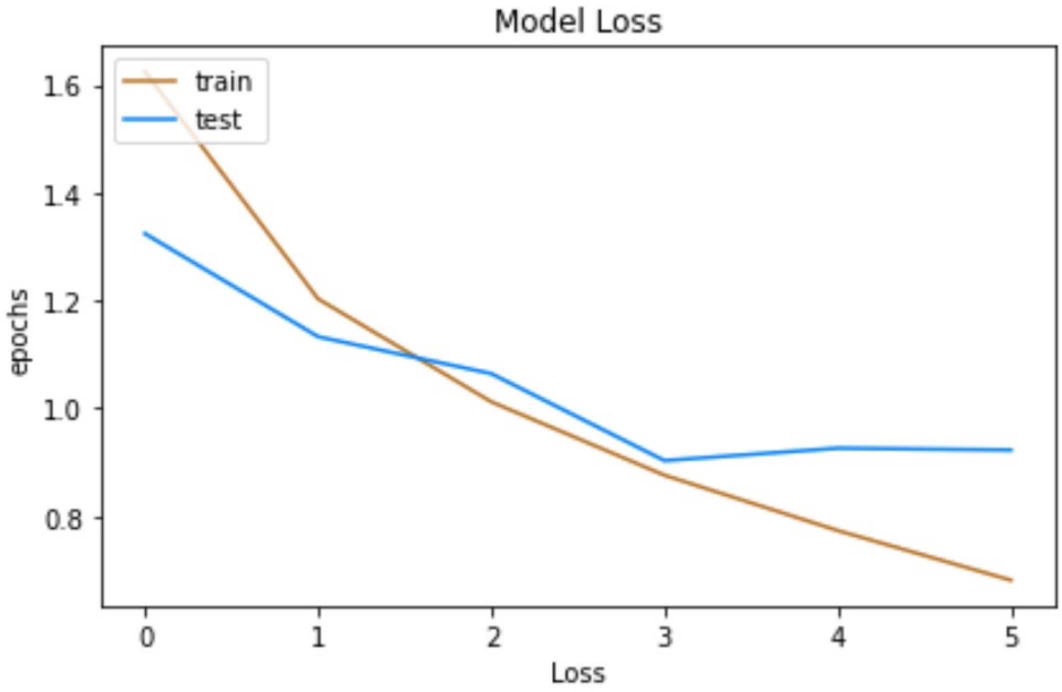
Model Accuracy with **ADAM** optimizers is around 73%

Model accuracy with **NAdam** Optimizers:



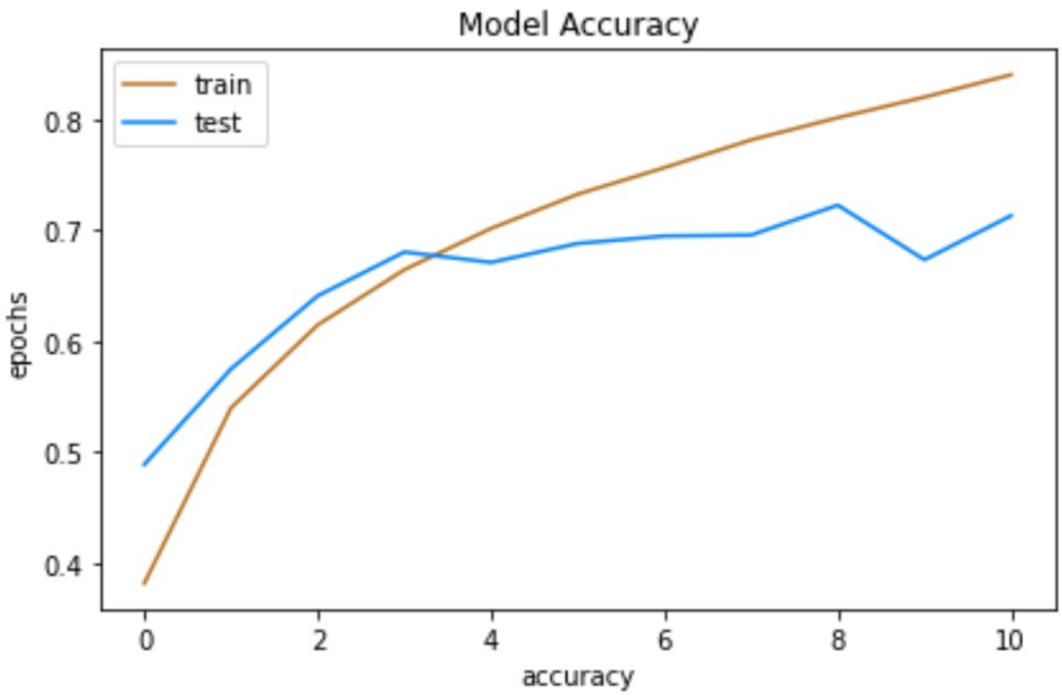
Model Loss with

NAdam Optimizers:



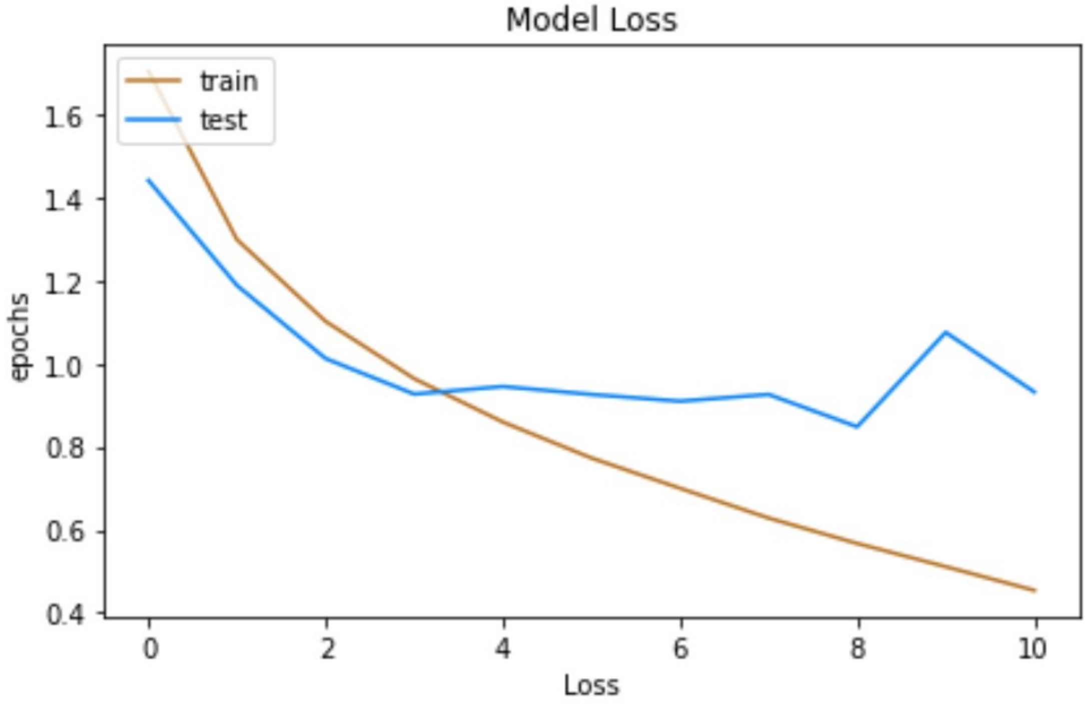
Model Accuracy with **NADAM** optimizers is around 68.24%

Model accuracy with **RMsprop** Optimizers:



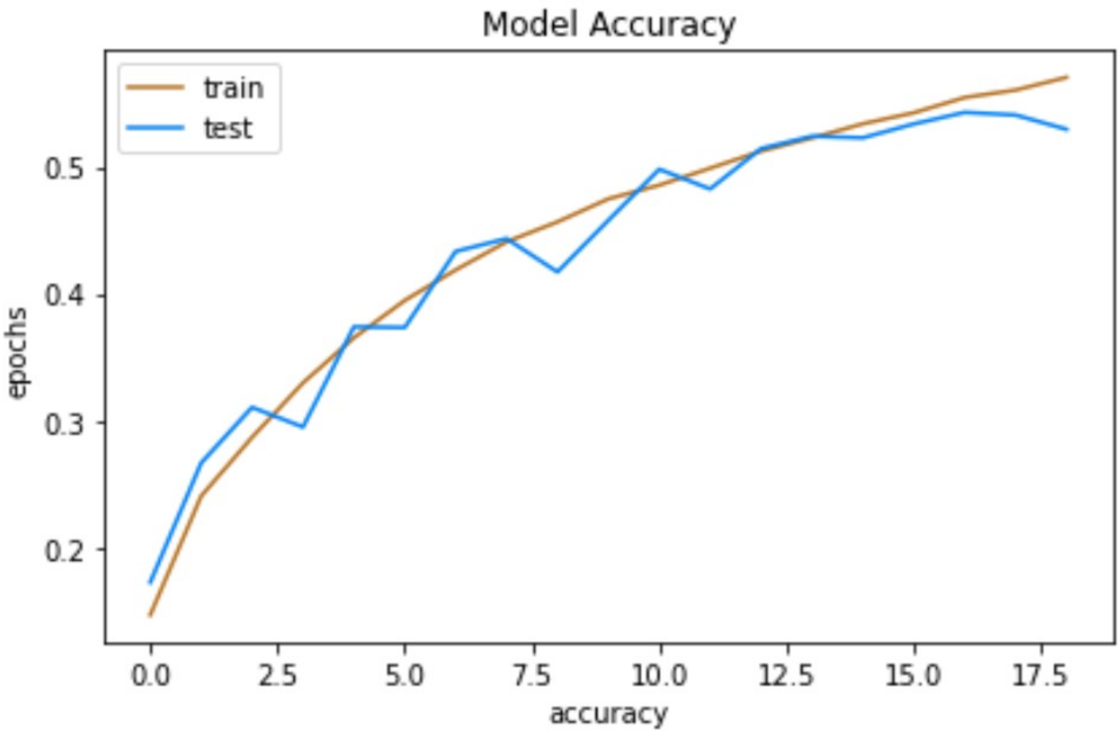
Model Loss with

RMsprop Optimizers:



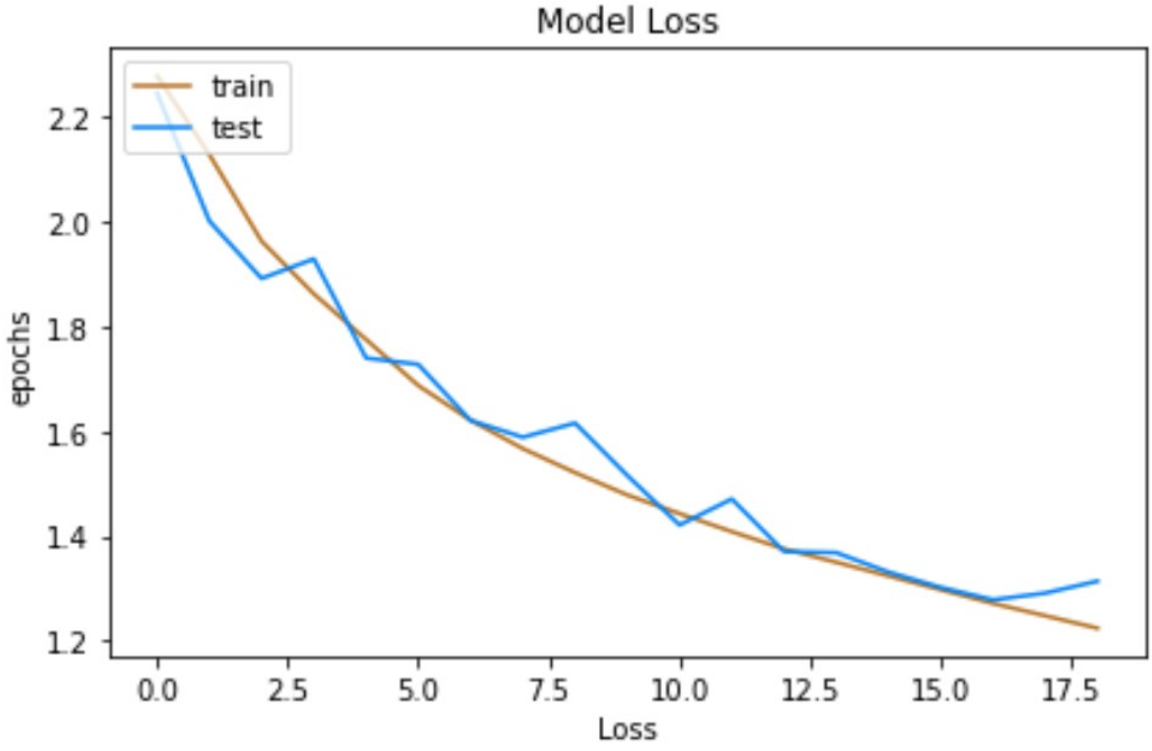
Model Accuracy with **RMsprop** optimizers is around 71%

Model accuracy with **SGD** Optimizers:



Model Loss with

SGD Optimizers:



Model Accuracy with **SGD** optimizers is around 53%

`Summary :`

Optimizers Name	Loss	Accuracy
1. Adam	.84	.73
2. NAdam	.92	.68
3. RmsProp	.93	.71
4. SGD	1.31	.53

We got different accuracy metrics for different optimizers using condition.