**Report :**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Model type | Training Accuracy | Validation Accuracy | Trainable Parameters | Tested by |
| 1. | NN (TF) | 95.41 | 67.46 | 0.12M | Aditya |
| 2. | NN (Keras) | 81.72 | 82.69 | 0.28M | Aditya |
| 3. | NN (TF) | 70.55 | 71.74 | 0.71M | Aditya |
| 4. | NN(TF) | 74.17 | 76.02 | 0.72M | Aditya |
| 5. | NN(TF) | 69.33 | 71.75 | 0.59M | Aditya |
| 6 | NN(keras) |  |  | 0.30M | Aditya |
|  |  |  |  |  |  |

Notes :

* Model 1 is the basic neural network model with 3 convolution layer and a simple neural net.
* Model 2 consists of more convolution layer along with addition of dropout layers.
* Model 3 is upgraded version of model 2, i.e. added another convolution layer + another dense layer in neural net.
* Model 4 inspired from NiN approach of having a mlpconvlayer instead of conv layer. This is similar as adding a Conv (1x1) layer; therefore, model 3 was upgraded. [5]
* Model 5 is same as model 4; only dropout layer is added in neural network layer.
* Model 6 is similar to model 2 as it has shown better performance so far. Therefore, in the model, I have added batch normalization and weight decay. Also, we are augmenting the images, to improve performance.

**References :**

1. <https://www.cs.toronto.edu/~kriz/cifar.html>
2. <https://github.com/toelt-llc/youtube/blob/master/First_Example_of_a_CNN_(CIFAR10).ipynb>
3. <https://machinelearningmastery.com/save-load-keras-deep-learning-models/>
4. <https://arxiv.org/pdf/1505.00853v2.pdf>
5. <https://arxiv.org/pdf/1312.4400v3.pdf>