Final accepted Result:

|  |  |  |  |
| --- | --- | --- | --- |
| Layer | Neurons | Activation Function | Other info |
| Input Layer | 32\*32\*3=3072 |  |  |
| Conv2D | 30\*30\*32=28,800 | Relu |  |
| Conv2D | 28\*28\*32=25,088 | Relu |  |
| Conv2D | 26\*26\*64=43264 | Relu |  |
| Conv2D | 24\*24\*64=36864 | Relu |  |
| Conv2D | 20\*20\*128=51200 | Relu |  |
| Conv2D | 16\*16\*128=32768 | Relu |  |
| Dense | 20 | Relu |  |
| Output | 10 | softmax |  |



After 10 epochs:

trainingloss: 0.1720 - trainingacc: 0.9425

val\_loss: 2.5947 - val\_acc: 0.5338

testingloss: 2.5740 - testingacc: 0.5354

After this it feels that if may happen that due to a greater number of convolutions of filter, we may have lost many features.



After 10 epochs:

loss: 0.1903 - acc: 0.9366 (Training)

val\_loss: 2.6047 - val\_acc: 0.5536 (Validation)

loss: 2.6217 - acc: 0.5458 (Testing)

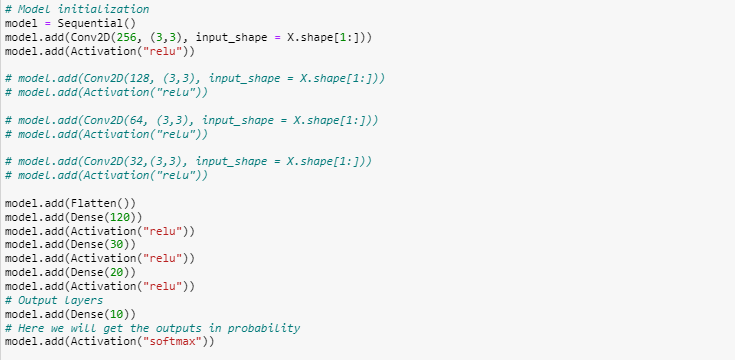


After 10 epochs:

loss: 0.7582 - acc: 0.7359 (Training)

val\_loss: 1.4369 - val\_acc 0.5539(Validation)

loss: 1.4228 - acc: 0.5520 (Testing)



loss: 0.4554 - acc: 0.8503 (Training)

val\_loss: 1.7626 - val\_acc: 0.5408 (Validation)

loss: 1.7725 - acc: 0.5298 (Testing)



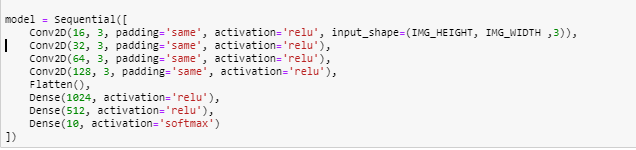
After 10 epochs:

loss: 0.1840 - acc: 0.9392 (Training)

val\_loss: 2.3805 - val\_acc: 0.5398 (Validation)

loss: 2.4350 - acc: 0.5379 (Testing)

New method with RGB:

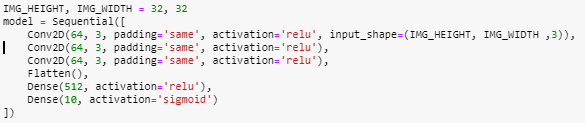


After 10 epochs:

loss: 0.0459 - acc: 0.9866 (Training)

val\_loss: 2.9333 - val\_acc: 0.6011 (Validation)

loss: 2.8199 - acc: 0.6050s (Testing)

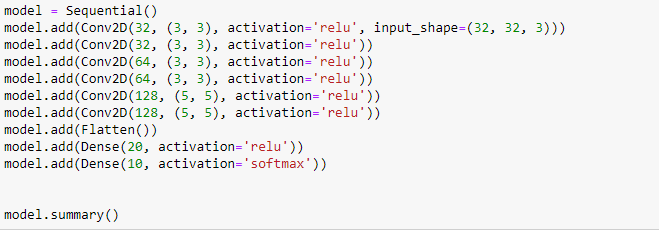


After 10 epochs:

loss: 0.0508 - acc: 0.9845 (Training)

val\_loss: 2.5727 - val\_acc: 0.6038 (Validation)

loss: 2.6422 - acc: 0.5984 (Testing)



This is the final approach that I have used for final submission.

After 10 epochs:

loss: 0.2244 - acc: 0.9199 (Training)

val\_loss: 1.3938 - val\_acc: 0.6711 (Validation)

loss: 1.3761 - acc: 0.6787 (Testing)

Making a dataset:

We have five files for the dataset where batches.meta gives the information about labels name and number of batches per class as well as the dimension of each image data. All other batch files give the dataset. Each batch files have 10000 image data. Each batch file contains data in the dictionary format. It contains data, labels, filenames etc.

Total number of classes is: 10 and classes names are: 'airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'truck'

To make the training data, we need to fetch the data and labels. So, we have taken the data from the batch files and made the training data.

First, we have tried with grayscale images and it doesn’t increase the accuracy, so later we have tried with RGB image.



Then we have try to do shuffle of the data randomly. After that we have divided the data into training, validation and testing data.

First, we have tried with grayscale images but the accuracy is so much limited around 50% accuracy. So, we have thought to use RGB images because it may happen that grayscale images can lost some useful information from the images. We have tried different combination of the Conv2D , Dense layer and finally we have attain the combination of the layers by which we got the accuracy of 67%.