ASSIGNMENT 1-Keras

Dhanashri Palodkar-1357556

Problem 1:

Using Keras, build a MLP to classify the CIFAR-10 dataset. Note that each record is of size 1*3072. Starting with the MNIST example code, build a MLP to classify the data into the 10 classes.

Model 1:

Using Keras to build a MLP for classifying the CIFAR-10 dataset. Each record is of size 1*3072. Building a MLP to classify the data into the 10 classes.

DATASET:

CIFAR-10 dataset The CIFAR-10 dataset consists of 60000 32x32 colour images in 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images.

The dataset is divided into five training batches and one test batch, each with 10000 images. The test batch contains exactly 1000 randomly-selected images from each class. The training batches contain the remaining images in random order, but some training batches may contain more images from one class than another. Between them, the training batches contain exactly 5000 images from each class.

http://www.cs.utoronto.ca/~kriz/cifar.html

Modify the following parameters and discuss the effect of changing parameters on loss and Accuracy:

Answer

	No of epochs	Batch size	Number of neurons	Number of layers	Learning rate	Activation functions	Dropout rates
Model 1	20	128	3072	8	0.0001	relu, softmax	0.2
Model 2	10	256	3072	10	0.0001	relu, softmax	0.2

- Based on the above results, **MLP model 1** performed well and gave an accuracy of **46.59%**
- MLP using relu & softmax activation functions improved the results.

Model Fitting:

Train on 50000 samples, validate on 10000 samples

Epoch 14/20
50000/50000 [==============================] - 36s 719us/step - loss: 1.5399 - acc: 0.4502 - val_loss: 1.5433 - val_acc: 0.4515
Epoch 15/20

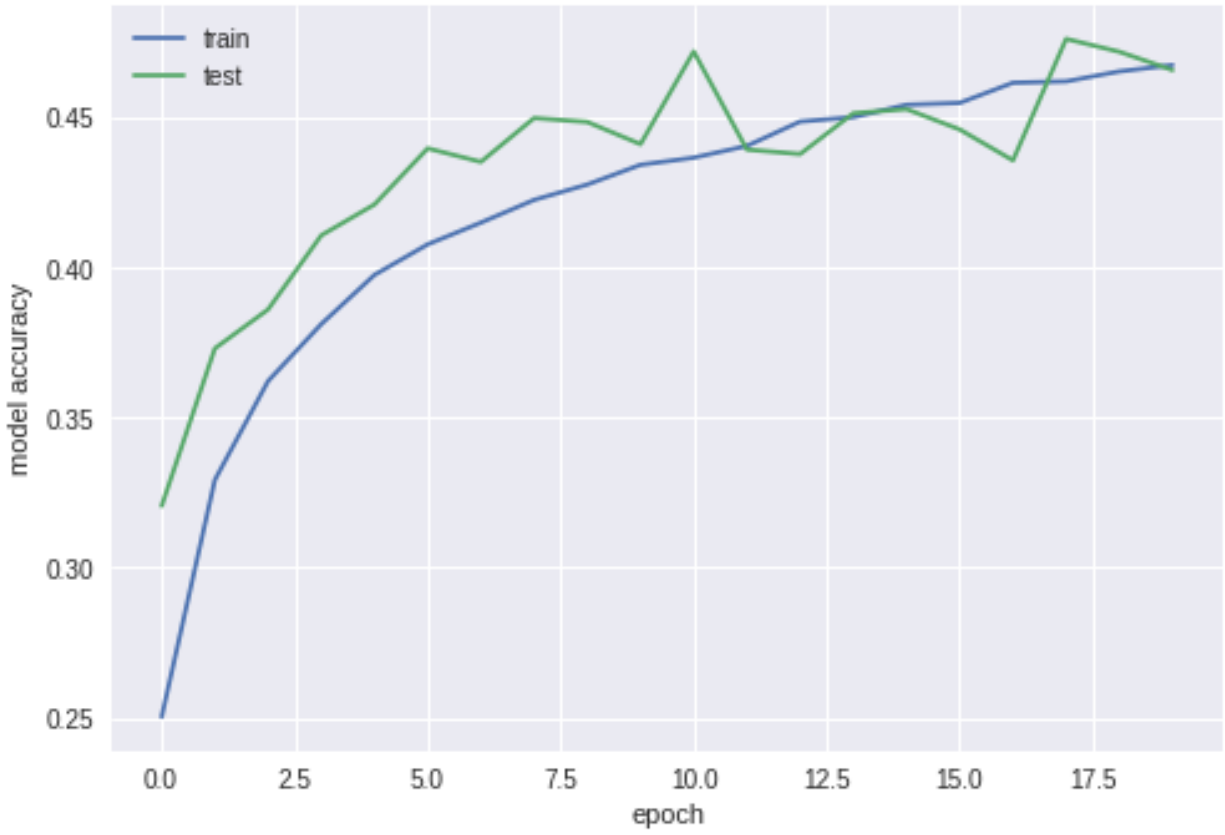
ASSIGNMENT 1-Keras

Dhanashri Palodkar-1357556

50000/50000 [==============================] - 39s 788us/step - loss: 1.5340 - acc: 0.4543 - val_loss: 1.5571 - val_acc: 0.4529
Epoch 16/20
50000/50000 [==============================] - 39s 789us/step - loss: 1.5290 - acc: 0.4550 - val_loss: 1.5415 - val_acc: 0.4461
Epoch 17/20
50000/50000 [==============================] - 39s 779us/step - loss: 1.5140 - acc: 0.4617 - val_loss: 1.5809 - val_acc: 0.4358
Epoch 18/20
50000/50000 [==============================] - 39s 781us/step - loss: 1.5136 - acc: 0.4621 - val_loss: 1.4962 - val_acc: 0.4763
Epoch 19/20
50000/50000 [==============================] - 40s 793us/step - loss: 1.5072 - acc: 0.4655 - val_loss: 1.4803 - val_acc: 0.4720
Epoch 20/20
50000/50000 [==============================] - 39s 787us/step - loss: 1.4982 - acc: 0.4676 - val_loss: 1.5025 - val_acc: 0.4659
Test loss: 1.5024641061782837
Test accuracy: 0.4659

Plots off Best Model

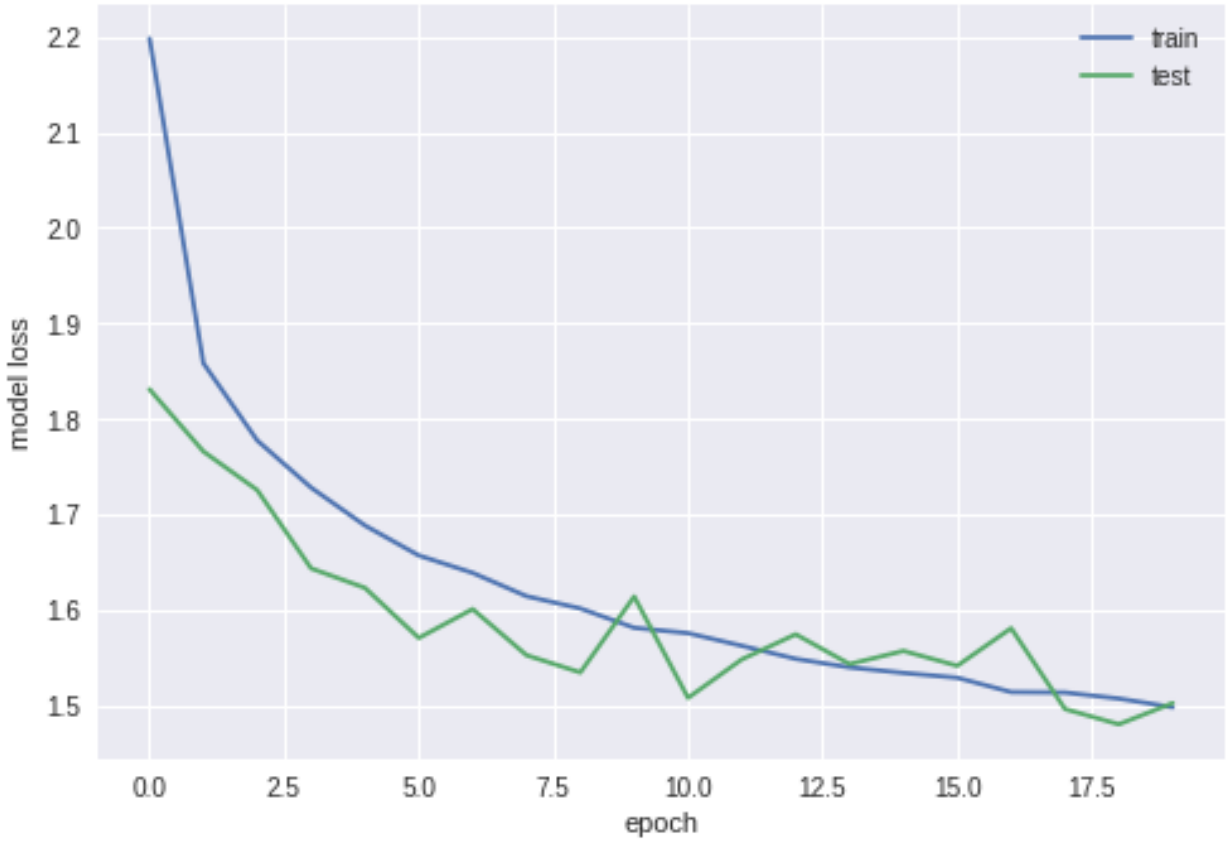
Model Accuracy Plot



ASSIGNMENT 1-Keras

Dhanashri Palodkar-1357556

Model Loss Plot



Results:

1) Provide a recommendation for the best model you would recommend for classification. Which model (with parameter values) would you choose and why?

According to me, the model that worked best for MLP was model 1. It had an accuracy of 46.59%.The best parameters that helped improve the model are:

- epochs = 20
- batch_size = 128
- keep_probability = 0.7
- learning_rate = 0.001
- Optimizer=rvmplot
- Lr=0.001

I will Choose Model 1 since the model 2 gave me accuracy of 25.69 which is less. When I changed the values I got better accuracy which is 46%

ASSIGNMENT 1-Keras

Dhanashri Palodkar-1357556

2) Comment on how good your model is ? Does it overfit/underfit data ? What could you do to improve the model?

I ran it for 80 epochs and got almost 46.59% accuracy Since 46.59% can not be considered as perfect accuracy. Since it is not capturing all the data, My Model underfits To improve the model the learning rate should be improvised apart from that number of neurons can be more complicated for a better fit. Along with these parameters, the number of epochs can also be increased for the model to train better.

Also I would like to include oulier and and add features.

Part 2

continue using the CIFAR-10 dataset for Part 2 but this time around, we will use Keras to build a CNN network

Modify the following parameters and discuss the effect of changing parameters on loss and Accuracy:

	No of epochs	Batch size	Number of neurons	Number of layers	Learning rate	Activation functions	Dropout rates
Model 1	15	128	7000	10	0.01	relu, softmax	0.25
Model 2	10	64	7000	15	0.01	relu, softmax	0.5

- Based on the above results**, CNN model 1** performed well and gave an accuracy of **71.86%**
- MLP using relu & softmax activation functions improved the results.

Model fitting

Train on 50000 samples, validate on 10000 samples
Epoch 10/15
50000/50000 [==============================] - 1283s 26ms/step - loss: 1.0174 - acc: 0.6422 - val_loss: 0.9622 - val_acc: 0.6633
Epoch 11/15
50000/50000 [==============================] - 1284s 26ms/step - loss: 0.9789 - acc: 0.6573 - val_loss: 0.9418 - val_acc: 0.6681
Epoch 12/15
50000/50000 [==============================] - 1287s 26ms/step - loss: 0.9424 - acc: 0.6710 - val_loss: 0.8736 - val_acc: 0.6980
Epoch 13/15
50000/50000 [==============================] - 1285s 26ms/step - loss: 0.9105 - acc: 0.6820 - val_loss: 0.8419 - val_acc: 0.7108
Epoch 14/15

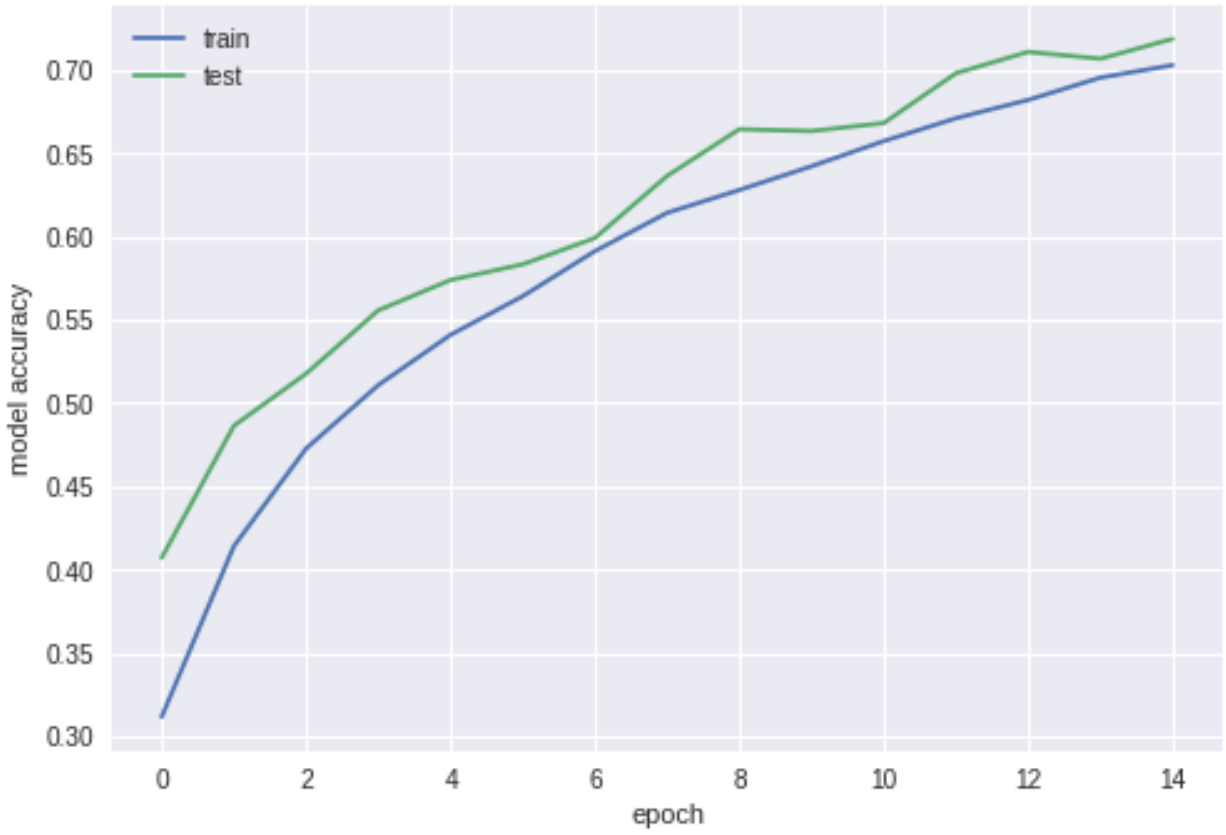
ASSIGNMENT 1-Keras

Dhanashri Palodkar-1357556

50000/50000 [==============================] - 1280s 26ms/step - loss: 0.8770 - acc: 0.6954 - val_loss: 0.8457 - val_acc: 0.7067
Epoch 15/15
50000/50000 [==============================] - 1280s 26ms/step - loss: 0.8523 - acc: 0.7028 - val_loss: 0.8087 - val_acc: 0.7186

Plots off Best Model

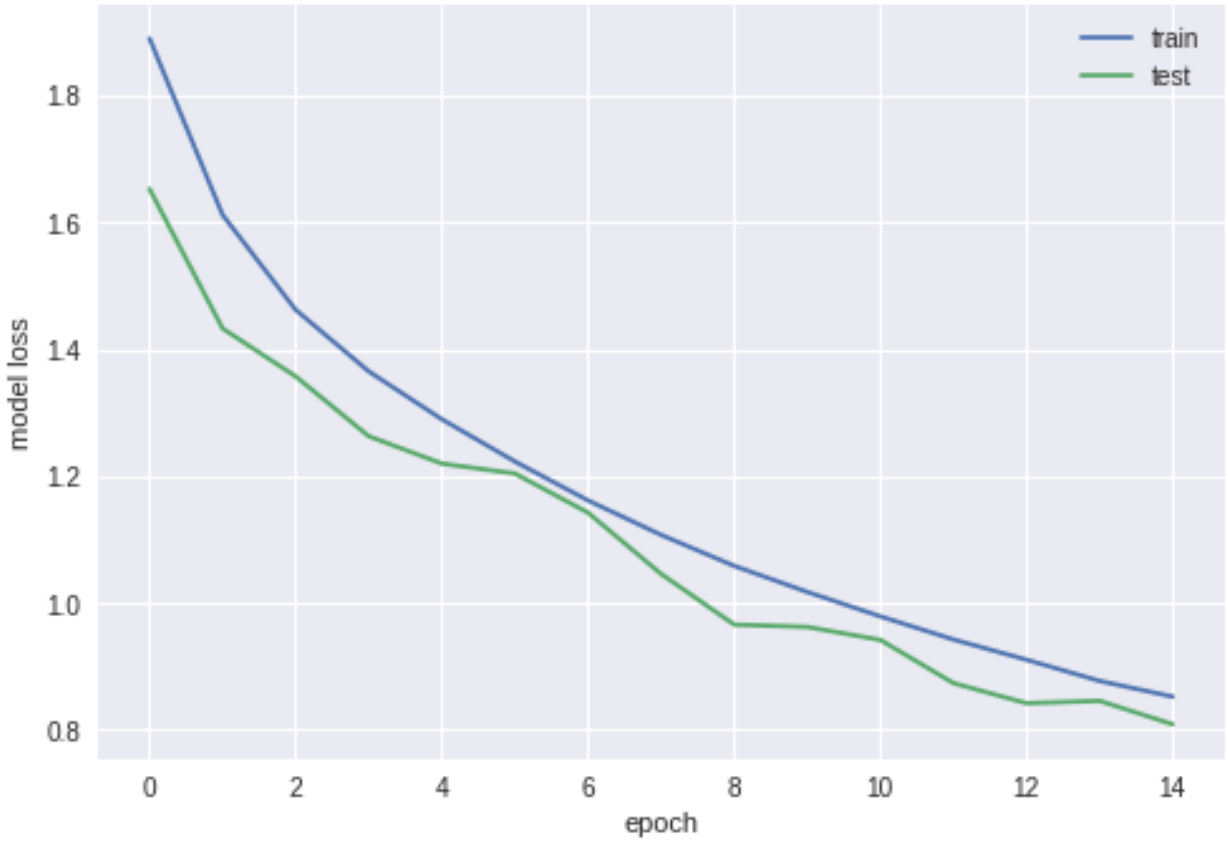
Model Accuracy Plot



ASSIGNMENT 1-Keras

Dhanashri Palodkar-1357556

Model Loss Plot



Results:

1) Provide a recommendation for the best model you would recommend for classification. Which model (with parameter values) would you choose and why?

According to me, the model that worked best for MLP was model 1. It had an accuracy of 71.86%.The best parameters that helped improve the model are:

- epochs = 15
- batch_size = 128
- Keep probability = 0.7
- learning_rate = 0.01

ASSIGNMENT 1-Keras

Dhanashri Palodkar-1357556

- Optimizer = opt

2) Comment on how good your model is ? Does it overfit/underfit data ? What could you do to improve the model?

I ran it for 90 epochs and got almost 71.86% accuracy. This model under fit the data. To improve the model the learning rate should be improved. Also, number of neurons can be more complicated for a better fit. Along with these parameters, the number of epochs can also be increased for the model to train better.