**Results**

Cifar-10 MLP

1. Test loss = 1.5600060455322267, Test accuracy = 0.4467
   1. batch\_size = 128
   2. num\_classes = 10
   3. epochs = 20
   4. Layers = 3
   5. Number of neurons in a layer = 512
   6. Activation = “relu”
   7. Dropout = 0.2
2. Test loss = 1.5545186204910277, Test accuracy = 0.4637
   1. batch\_size = 128
   2. num\_classes = 10
   3. epochs = 10
   4. Layers = 3
   5. Number of neurons in a layer = 512
   6. Activation = “relu”
   7. Dropout = 0.2
3. Test loss: 1.691791695022583, Test accuracy: 0.3898
   1. batch\_size = 64
   2. num\_classes = 10
   3. epochs = 10
   4. Layers = 3
   5. Number of neurons in a layer = 512
   6. Activation = “relu”
   7. Dropout = 0.2
4. Test loss: 1.6212852376937865, Test accuracy: 0.415
   1. batch\_size = 64
   2. num\_classes = 10
   3. epochs = 10
   4. Layers = 3
   5. Number of neurons in a layer = 256
   6. Activation = “relu”
   7. Dropout = 0.2
5. Test loss: 1.7278007930755614, Test accuracy: 0.4009
   1. batch\_size = 64
   2. num\_classes = 10
   3. epochs = 10
   4. Layers = 5
   5. Number of neurons in a layer = 256
   6. Activation = “relu”
   7. Dropout = 0.2
6. Test loss: 1.7867918239593505, Test accuracy: 0.3629
   1. batch\_size = 64
   2. num\_classes = 10
   3. epochs = 10
   4. Layers = 5
   5. Number of neurons in a layer = 256
   6. Activation = “tanh”
   7. Dropout = 0.2
7. Test loss: 1.8273195960998536, Test accuracy: 0.3907
   1. batch\_size = 64
   2. num\_classes = 10
   3. epochs = 10
   4. Layers = 5
   5. Number of neurons in a layer = 256
   6. Activation = “elu”
   7. Dropout = 0.2
8. Test loss: 1.7028786176681519, Test accuracy: 0.3912
   1. batch\_size = 64
   2. num\_classes = 10
   3. epochs = 10
   4. Layers = 5
   5. Number of neurons in a layer = 256
   6. Activation = “elu”
   7. Dropout = 0.3

Cifar-10 CNN

1. Test loss: 1.011834213066101, Test accuracy: 0.6388
   1. batch\_size = 32
   2. num\_classes = 10
   3. epochs = 10
   4. Layers = 3
   5. Number of neurons in a layer = 512
   6. Activation = “relu”
   7. Dropout = 0.25
   8. Learning rate = 0.0001
2. Test loss =1.6457371658325195, Test accuracy = 0.4157
   1. batch\_size = 32
   2. num\_classes = 10
   3. epochs = 5
   4. Layers = 3
   5. Number of neurons in a layer = 512
   6. Activation = “relu”
   7. Dropout = 0.25
   8. Learning rate = 0.0001
3. Test loss: 0.9773193073272705, Test accuracy: 0.6669
   1. batch\_size = 64
   2. num\_classes = 10
   3. epochs = 5
   4. Layers = 3
   5. Number of neurons in a layer = 512
   6. Activation = “relu”
   7. Dropout = 0.25
4. Test loss: 1.1862906442642212, Test accuracy: 0.5815
   1. batch\_size = 64
   2. num\_classes = 10
   3. epochs = 5
   4. Layers = 3
   5. Number of neurons in a layer = 512
   6. Activation = “elu”
   7. Dropout = 0.25
   8. Learning rate = 0.0001
5. Test loss: 0.8766417194366455, Test accuracy: 0.7008
   1. batch\_size = 32
   2. num\_classes = 10
   3. epochs = 5
   4. Layers = 3
   5. Number of neurons in a layer = 512
   6. Activation = “elu”
   7. Dropout = 0.25
   8. Learning rate = 0.0002

Cifar-10 CNN

Test loss: 0.8766417194366455, Test accuracy: 0.7008, 596s 381ms/step

Cifar-10 CNN\_GPU

Test loss: 1.0398049556732178, Test accuracy: 0.6346, 72s 46ms/step

**Analysis**

1. No of epochs:

With the increasing of the number, the loss will decrease, and the accuracy will increase.

2. Batch size:

With the increasing of batch size, the loss will decrease, and the accuracy will increase, except for over-fitting batch size

3. number of neurons in a layer

With the increasing of the number in hidden layer, the loss will increase, and the accuracy will decrease, but in order to avoid under-fitting, we cannot over-reduce the neurons.

4. numbers of layers

With the increasing of layers, the loss will increase, and the accuracy will decrease.

5. learning rate

With the increasing of learning rate, the loss will increase, and the accuracy will decrease.

6. activation function

I have tried three activation functions. Overall, ELU is the best activation function for both MLP and CNN model.

7. dropout rates

Dropout rate helps avoid over-fitting for other functions. With the increasing of dropout rate, the loss will increase, and the accuracy will decrease.

Thus, based on the results I have got, I would recommend the second model I used since it has the highest accuracy. However, my models seemed not very satisfied since none of has 50% accuracy in the end. I think it is due to underfitting. I think MLP probably not the best model for doing Cifar-10.