

# Experiment Observations and Conclusions

CIFAR 10 Image Data Multi-layer Perceptron Classifier Experiments

- The hyper parameters I will recommend for this model are:

activation	optimizer	num_layers	num_neurons	dropout_rate	batch_size	epochs
relu	Adam with learning rate @ 0.0003	2	256	0.2	128	20
relu	Adam with learning rate @ 0.0003	2	256	0.2	32	100
relu	SGD with learning rate @ 0.1	2	256	0.2	128	20
relu	SGD with learning rate @ 0.03	2	256	0.2	128	20
softplus	Adam with learning rate @ 0.0003	2	256	0.2	128	20
relu	Momentum with learning rate @ 0.003	2	256	0.2	128	20
relu	RMSprop with learning rate @ 0.0003	2	256	0.2	128	20
relu	Adam with learning rate @ 0.0001	2	256	0.2	128	20
relu	Momentum with learning rate @ 0.01	2	256	0.2	128	20
relu	Adam with learning rate @ 0.0003	2	256	0.2	256	100
relu	Adadelta with learning rate @ 0.1	2	256	0.2	128	20

Those are best models from the experiment, which reach an accuracy of 0.50~0.51. Below are some observations and summary from this experiment:

1. batch\_size = 128 (accuracy reaches to higher level, and the overall speed did not reduce too much)
2. dropout\_rate = 0.2 (0.4 and 0.6 are too big in this case - not a big amount of neurons - that model did not learn that much)
3. epochs = 100 (around 50 epoch, model seems to converge, set it to 100 epoch to see if there are further improvement, this will allow us to react more flexible to the training situation)
4. activation function @ hidden layer = 'relu' (faster and higher accuracy compared with other activation functions)
5. activation function @ output layer = 'softmax' (because it is a multi-class classification problem)
6. optimizer and learning rate : there's a couple of different combinations of the two which performed similarly: Adam with learning rate @ 0.003/0.001, SGD with learning rate @ 0.03/0.1, etc. worked really well

- **The configuration I will recommend for this model is:**

1. number of layers: 2 or 3 dense layers are fine (there's no need to further increase layers as no improvement on accuracy and loss has been observed)
2. number of neurons: 256 neurons is good choice (there's no need to further increase neurons as no improvement on accuracy and loss has been observed)

- **Comment on this Model:**

This model is not that good based on current accuracy level (0.51). It is under-fit overall. However given it is a shallow neural network with only 256 neurons per layers, this neural network is not bad.

- **Suggestions on this Model:**

To further improve this model, there are several steps we could take:

1. try to get more data to strengthen training data size and help with model generalization
2. use data augmentation to increase training data size and data variance
3. apply regularization method to help with over-fitting
4. try to implement different model - CNN, which is better tool for computer vision problem and helps with the need for invariance
5. try to use pre-trained layers and then fine-tune the last few layers to improve accuracy