Auto encoder Experiments for class data set.
Classes - 6
Hidden Layers - 3

Part 1: Vary the number of hidden units exponentially as follows; if your last name starts with a letter between A and K vary the number of neurons between 32, 64, 128, 256, and 512; in three layers only, for a total of 10 experiments. If your last name has any other letter (L to Z), you will do two layers and will vary the number of neurons between 32, 64, 128, 256, 512, 1024; for a total of 15 experiments.

For the experiments, I have performed following combinations and their respective results are -

Layer1	Layer2	Layer3	Error
512	32	128	0.8167
64	32	32	0.8
256	256	64	0.7833
64	256	64	0.7833
64	256	32	0.7833
64	128	32	0.7833
512	128	128	0.7667
512	128	32	0.7667
512	64	128	0.7667
512	32	64	0.7667
256	128	128	0.7667
256	64	32	0.7667
256	32	32	0.7667
64	128	64	0.7667
64	64	32	0.7667
512	256	32	0.75
512	128	64	0.75
512	64	64	0.75
512	32	32	0.75
256	128	64	0.75
256	64	64	0.75
128	256	32	0.75
128	32	64	0.75

512	256	128	0.7333
256	256	32	0.7333
128	64	128	0.7333
128	64	32	0.7333
128	**32**	**32**	**0.7333**
512	**256**	**64**	**0.7167**
512	64	32	0.7167
256	**128**	**32**	**0.7167**
256	32	128	0.7167
256	32	64	0.7167
64	128	128	0.7167
256	**256**	**128**	**0.7**
256	64	128	0.7
128	256	64	0.7
128	**128**	**32**	**0.7**
128	**64**	**64**	**0.7**
128	**32**	**128**	**0.7**
64	64	128	0.7
128	**128**	**128**	**0.6833**
64	32	128	0.6833
64	32	64	0.6833
64	256	128	0.6667
64	**64**	**64**	**0.6667**
128	256	128	0.65
128	**128**	**64**	**0.65**

The table highlights 10 results that are considerable for changing the learning rates of the experiments.

Which combination of neurons gave you the best testing accuracy? Why do you think that is?

Answer:
From the experiments, that have architecture like (512,32,128) has the highest error rate of nearly 80% which looks like the case of underfitting. Since the data is too limited and number of

neurons in hidden layers are unbalanced, the network is not able to perform feature extraction at its best.
Where in the architecture like (64,64,64) has much better result(error 66%) because of balanced network architecture.
From the results, 128,128,64(layer 1,layer 2, layer 3) architecture gives us the least result of 65% because of network structure.
Again, the results are not that good because of quality of the data and images. The images used for training are all taken from different devices and lightning conditions, object closeness and image quality are the major factors that contribute towards the error rate of network.

From the above experiments for part 2, I worked with architecture of Layers - 128,128,64 which gave the least error from the list.
After performing experiments with different learning rates, results are -

Learning Rate	Layer 1	Layer 2	Layer 3	Error
0.01	**128**	**128**	**64**	0.65
0.001	**128**	**128**	**64**	0.7
0.0001	**128**	**128**	**64**	
0.1	**128**	**128**	**64**	
0.01	**256**	**128**	**32**	
0.001	**256**	**128**	**32**	
0.0001	**256**	**128**	**32**	
0.1	**256**	**128**	**32**	

The blank spaces in table below has the program running and I will update the same on git as soon as they are over.

Which combination of neurons gave you the best testing accuracy? Why do you think that is?

Answer-
Till now, the change of learning rate to 0.01 in the first result gives the least error rate. As we go to learning rate 0.001, the error increases since the network needs more time to converge and hence it produces more error.