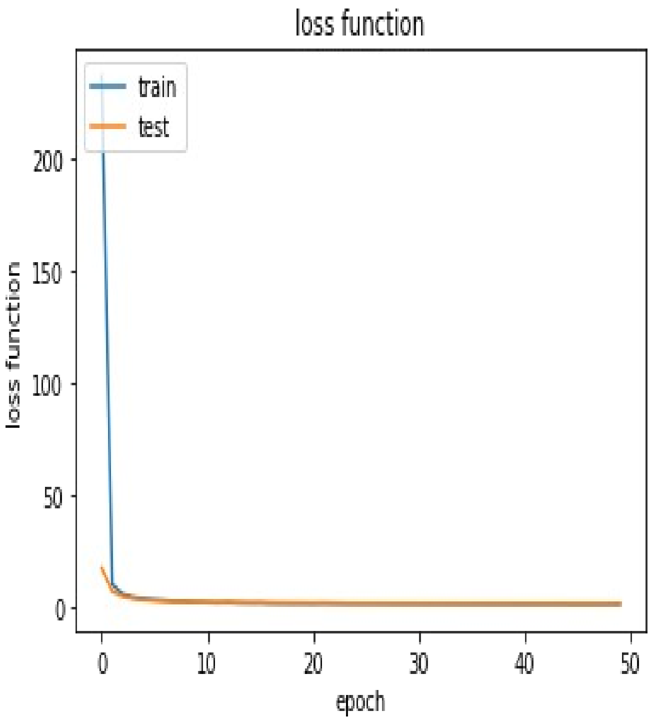
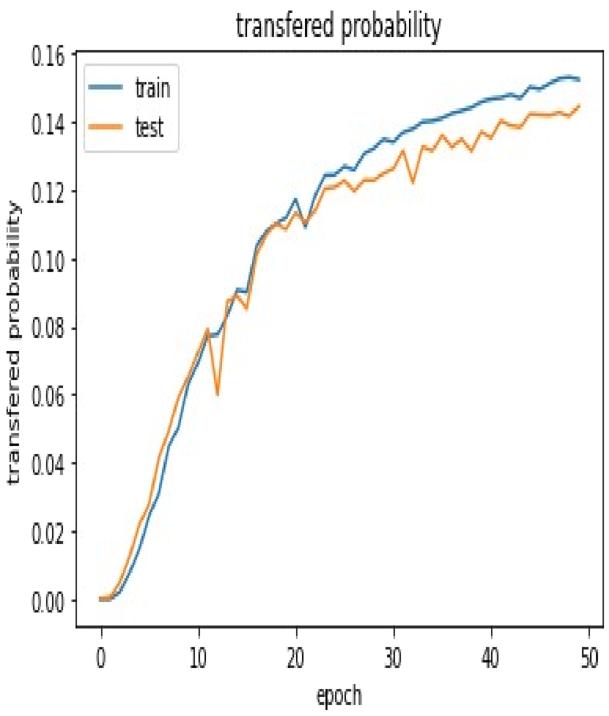
Math 6373 HW2

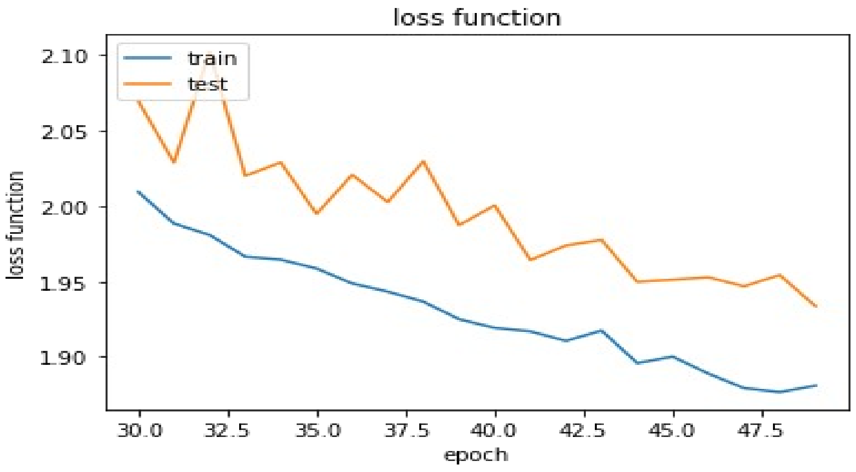
Dataset: CIFAR10 dataset
Training set has 50000 cases. Test set has 10000 cases.
Each case is an image of size (32,32,3), we transfer every image to a gray scale image of size (32,32) and reshape every image to a long vector of length 1024.
There are 10 classes in this data set: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck.

Q1
hidden layer size h=1024
Since the training progresses very slowly for this dataset, I plot the average cross entropy and the transformed average cross entropy in the following two figures for the first 50 epochs as the early training.

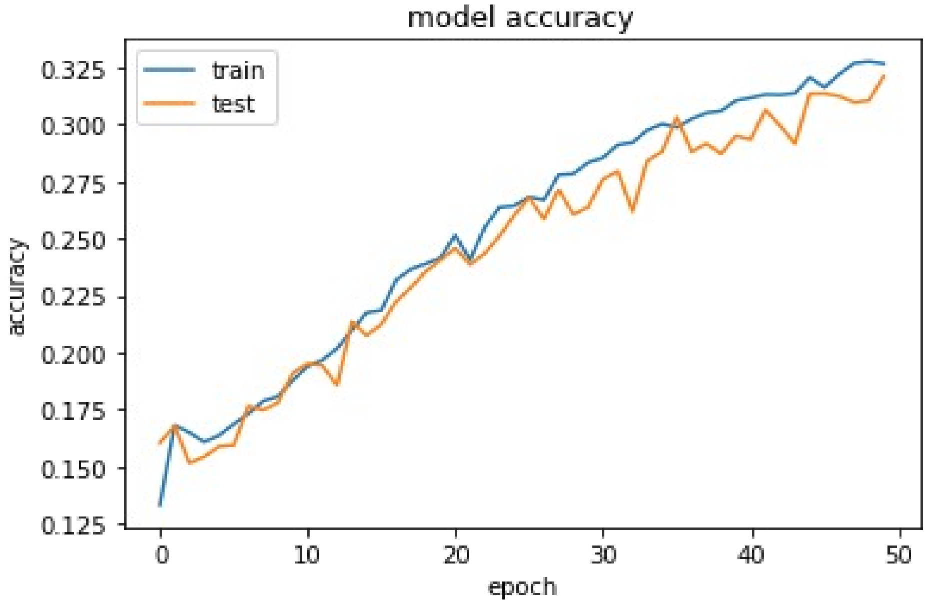




The loss function curve(average cross entropy): we see that the average cross entropy on training set and test set both decreases while training. However, if we plot the last 20 epochs in the following figure, we see that the training has not stabilized yet.



The transformed estimated probability curve: these two transformed probability curves shows that the average accuracy (estimated by the curves) on the training set and test set is increasing while training and the training has not stabilized yet. The true accuracy curve of the training set and test is in the following figure. We see that the training has not stabilized yet at 50 epochs. The accuracy on the training set is 0.327, the accuracy on the test set is 0.321.

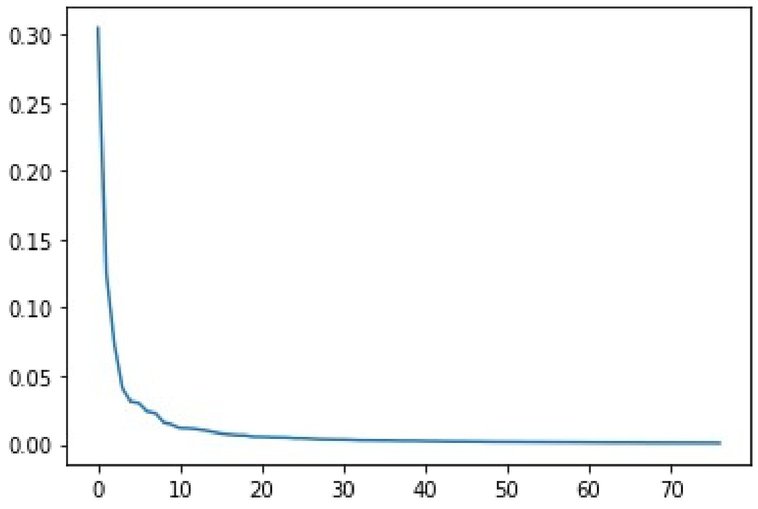


AVCRE(class Clj) for j=1,2,…,10:
13.214936, 11.167422, 13.479081, 13.080796, 13.717084 12.709615, 13.323832, 12.515389, 12.029563, 11.629509.

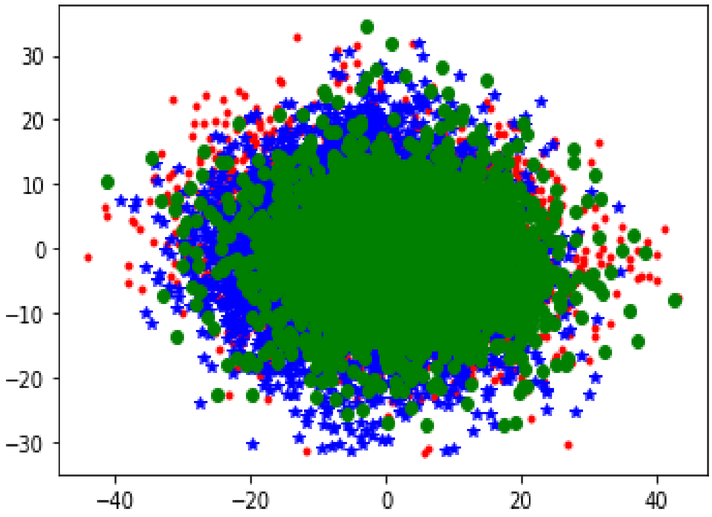
the transformed probability of these 10 values:
1.8231657e-06 1.4127006e-05 1.39994e-06 2.0848863e-06 1.1034334e-06
3.0219303e-06 1.6350592e-06 3.6697413e-06 5.96523e-06 8.8995575e-06

The accuracy on predicting each class is extremely low. At epoch 50, class 2 and class 10 have relatively higher accuracy.

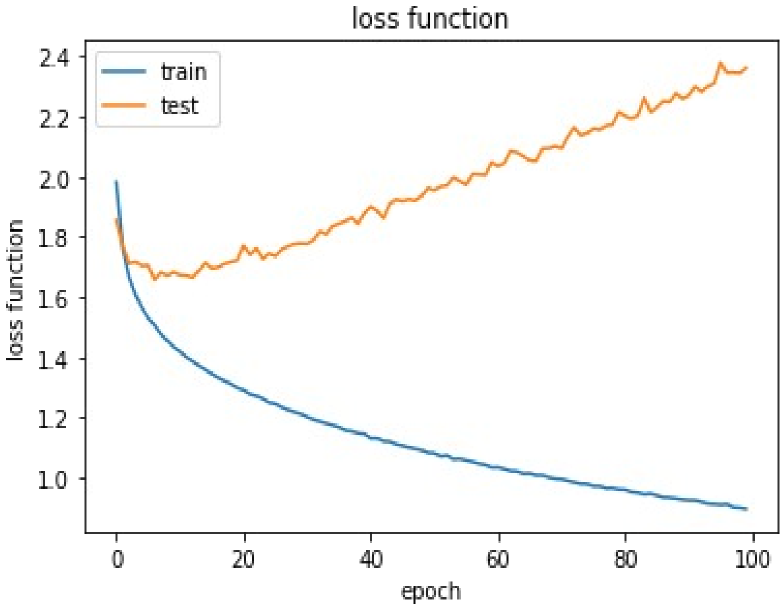
Q2:
90% PEV curve



We plot the class 1(red) data, class2(blue) data, class3(green) data by first 2 components after PCA in the following figure. We see that it is still quite difficult to classify even the first 3 classes after PCA



h_low=79. We plot the average cross entropy curve in the following figure. We see that overfitting happens around 20 epochs. The training on the training set has not stabilized after 100 epochs. The safe zone is from 5epoch to 15 epoch. The minimum loss on the test set happens at 8 epoch, so m*=8.



Confusion matrix on training set

```
[0.5376 0.0318 0.077 0.0296 0.0732 0.0214 0.029 0.0342 0.1126 0.033 ]
[0.0382 0.7074 0.0242 0.0342 0.023 0.028 0.0434 0.0224 0.056 0.1612]
[0.0978 0.0074 0.4158 0.0936 0.1324 0.1024 0.049 0.0466 0.0242 0.0116]
[0.028 0.0128 0.0762 0.3946 0.0602 0.188 0.0734 0.053 0.0222 0.022 ]
[0.0712 0.0122 0.151 0.0736 0.4616 0.0604 0.0788 0.0516 0.028 0.0144]
[0.0118 0.0042 0.0522 0.1126 0.0268 0.384 0.025 0.0426 0.0106 0.0108]
[0.0256 0.025 0.0778 0.1 0.0754 0.0642 0.6222 0.0196 0.0202 0.0242]
[0.0298 0.0176 0.0658 0.0828 0.0896 0.1062 0.0274 0.6912 0.0118 0.033 ]
[0.14 0.0848 0.037 0.0402 0.0416 0.0232 0.026 0.02 0.6832 0.0946]
```

```
[0.02 0.0968 0.023 0.0388 0.0162 0.0222 0.0258 0.0188 0.0312 0.5952]
```

We see that the diagonal elements of this confusion matrix takes the largest value on each row, which means that more cases were correctly classified rather than misclassified.

Confusion matrix on test set

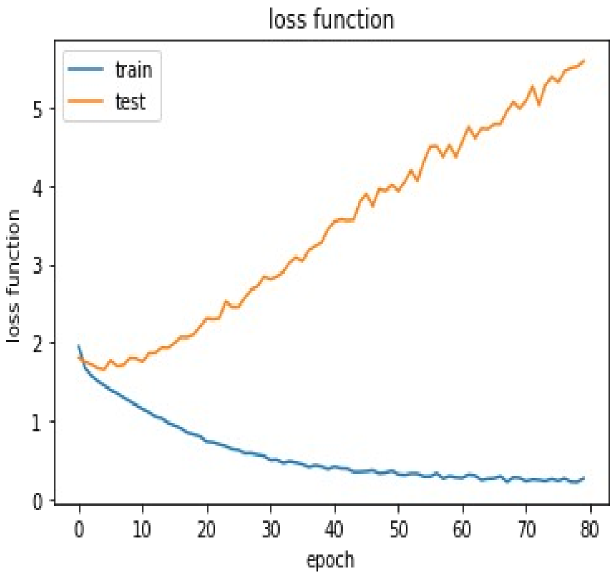
```
[0.447 0.047 0.08 0.057 0.079 0.033 0.045 0.042 0.126 0.048]
[0.043 0.543 0.025 0.047 0.046 0.024 0.07 0.026 0.078 0.192]
[0.113 0.009 0.329 0.094 0.129 0.119 0.064 0.052 0.03 0.016]
[0.027 0.025 0.092 0.279 0.074 0.199 0.093 0.075 0.038 0.043]
[0.076 0.02 0.162 0.08 0.325 0.078 0.115 0.094 0.036 0.023]
[0.013 0.005 0.063 0.12 0.036 0.298 0.036 0.071 0.014 0.016]
[0.045 0.055 0.087 0.135 0.114 0.076 0.472 0.045 0.023 0.03 ]
[0.034 0.023 0.089 0.097 0.123 0.109 0.044 0.518 0.027 0.064]
[0.164 0.124 0.048 0.037 0.048 0.029 0.031 0.038 0.589 0.105]
[0.038 0.149 0.025 0.054 0.026 0.035 0.03 0.039 0.039 0.463]
```

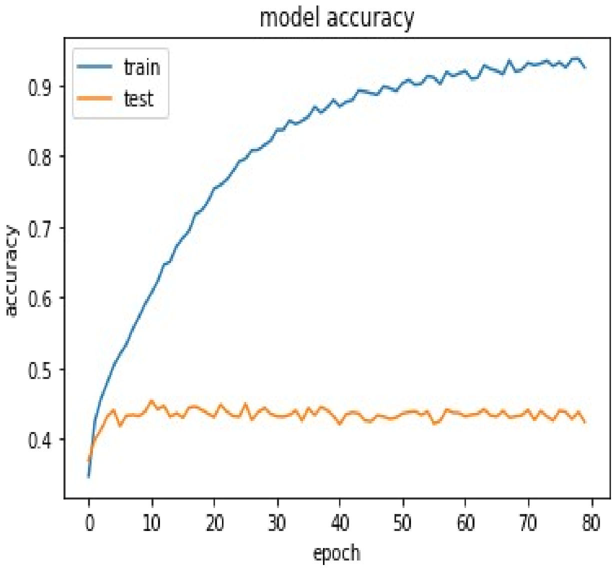
The diagonal values of confusion matrix on test set are still largest values of each row. These diagonal values take smaller values than the diagonal values of the confusion matrix on the training set (to be more convincing, we should compute the confidence interval of each diagonal value). So the MLPlow performs much better on the training set than on the test set.

The accuracy on the whole training set is 0.48 and the accuracy on the whole test set is 0.42.
I also tried the 99% PCA MLPlow, the result is similar to this.

Q3:
h_high=60+91+64+63+77+69+102+90+59+101=776

We plot the loss function and accuracy in the following figures.



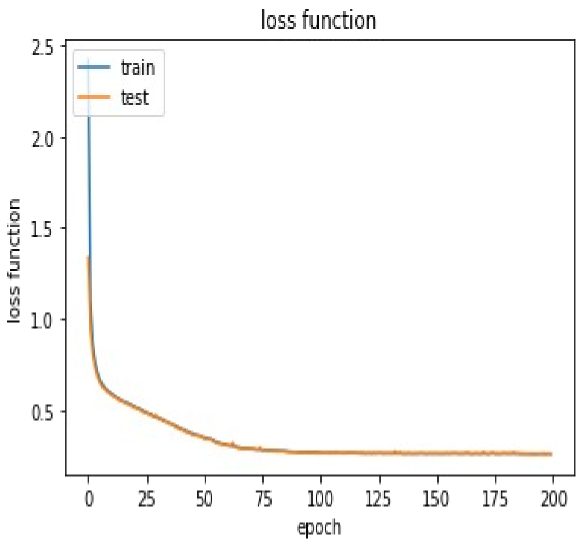


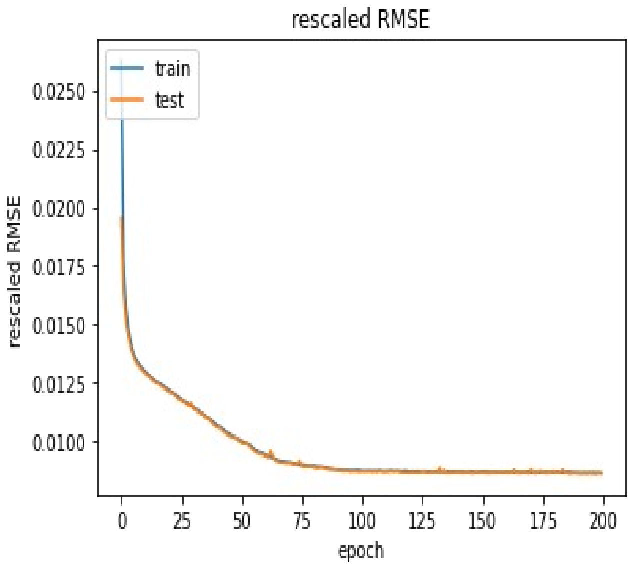
The overfitting happens around 10 epochs and the training on the training set has not stabilized after 80 epochs. The safe zone is from epoch 2 to 10 epoch. The minimum loss on test set happens at epoch 5.
The accuracy on the whole training set is 0.57(confidence interval: [0.5451, 0.5549]) and the accuracy on the whole test set is 0.44 (confidence interval [0.4351, 0.4449]). We see that this MLPhigh performs a little better than the MLPlow.

Q4:

95% linear autoencoder:rescaled RMSE: RMSE/mean of |Zn|=0.2016
95% MLP autoencoder: hidden layer size:461

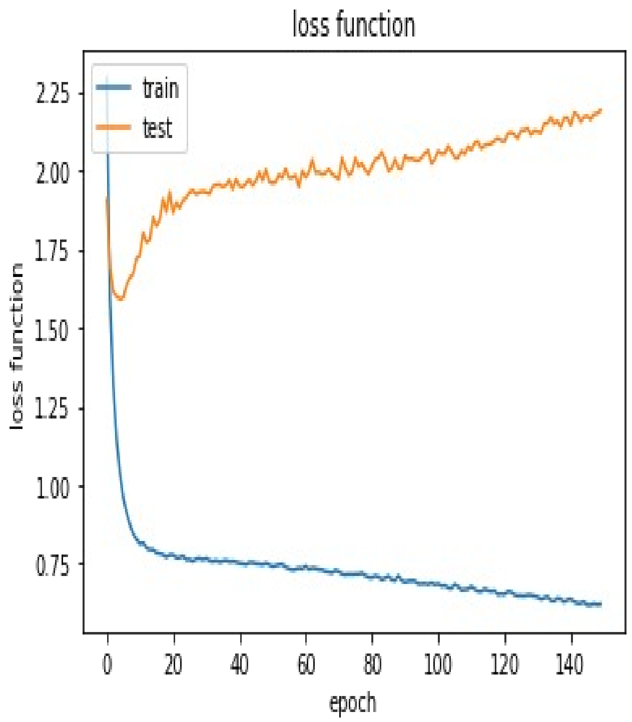
We plot the MSE and the rescaled RMSE on training set and test in the following two figures.

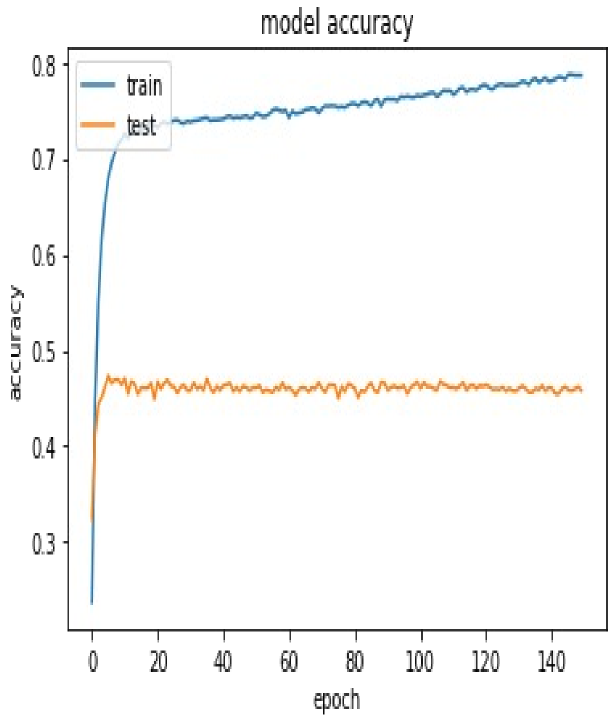




We see that the training on the training set stabilizes around 100 epochs. However, the loss on the test set stays lower than the loss on the training set almost the whole time. So we select the stopping epoch to be the minimum loss value on the test set after 100 epochs. m*=197. The re-scaled RMSE of MLP autoencoder is around 0.0086. So the MLP autoencoder performs better than the linear autoencoder.

Q5:
We plot the loss function and the accuracy on the training set and test set in the following two figures.





We select the stopping epoch same as before m*=7. Accuracy on the training set is 0.67. The confusion matrix on training set is

```
[[0.6532 0.0236 0.0666 0.0254 0.0392 0.0218 0.0112 0.014 0.0908 0.031 ]
[0.0146 0.7946 0.0066 0.0104 0.0066 0.008 0.0112 0.0024 0.0406 0.0858]
[0.0582 0.003 0.4976 0.0548 0.0642 0.07 0.0282 0.0178 0.019 0.009 ]
[0.033 0.0198 0.1112 0.6374 0.0516 0.2358 0.0784 0.0494 0.0244 0.0372]
[0.0908 0.0208 0.1728 0.0786 0.7146 0.0772 0.0672 0.071 0.0414 0.0174]
[0.008 0.002 0.0322 0.0442 0.0118 0.4466 0.0092 0.018 0.004 0.0106]
[0.0178 0.0264 0.0502 0.0706 0.041 0.0448 0.7692 0.0104 0.0156 0.0218]
[0.0152 0.0132 0.0306 0.0486 0.0506 0.0804 0.0102 0.8082 0.0074 0.0262]
[0.0982 0.0604 0.0222 0.0198 0.0156 0.0084 0.0112 0.0042 0.7404 0.0594]
[0.011 0.0362 0.01 0.0102 0.0048 0.007 0.004 0.0046 0.0164 0.7016]]
```

Accuracy on the test set is 0.46. The confusion matrix on the test set is

```
[[0.508 0.04 0.083 0.05 0.054 0.03 0.03 0.034 0.148 0.051]
[0.027 0.552 0.014 0.016 0.012 0.007 0.041 0.012 0.074 0.185]
[0.082 0.011 0.321 0.076 0.097 0.087 0.065 0.046 0.022 0.019]
[0.038 0.044 0.136 0.378 0.102 0.264 0.1 0.117 0.036 0.066]
[0.101 0.037 0.198 0.118 0.478 0.118 0.118 0.11 0.056 0.031]
[0.005 0.003 0.049 0.089 0.032 0.265 0.032 0.036 0.012 0.016]
[0.036 0.067 0.085 0.126 0.096 0.076 0.541 0.026 0.03 0.049]
[0.027 0.023 0.069 0.086 0.09 0.116 0.033 0.583 0.019 0.064]
[0.151 0.104 0.033 0.021 0.031 0.022 0.023 0.015 0.575 0.096]
 [0.025 0.119 0.012 0.04 0.008 0.015 0.017 0.021 0.028 0.423]]
```

Q6

	m*	trainacc	testacc	# hidden neuron	# param	CPU time(until m*)
MLPfirst		0.33	0.32	1024	1059850	420 sec
MLPlow	8	0.48	0.42	79	81775	10 sec
MLPhigh	5	0.57	0.44	776	803170	41 sec
MLPlong	7	0.67	0.46	461	82610	10 sec

MLPlong performs the best among these 4 models (compute the confidence interval of the accuracy). But MLPlong still has a low accuracy on the test set. So we implement the multiple MLP (10 MLPs) with MLP long architecture. The multiple MLP classifier has accuracy 0.47 on the test set.