Q1/ $p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k)$ since $\Sigma_k = \Sigma$

E-step:

$$\gamma_k = p(z=k|x) = \frac{p(z=k)\,p(x|z=k)}{p(x)} \quad \text{Bayes rule.}$$

$$= \frac{\pi_k \mathcal{N}(x|\mu_k, \Sigma)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x|\mu_j, \Sigma)} = \gamma_k.$$

$$\gamma_k = \frac{\pi_k \cdot \frac{1}{2\pi^{d/2}|\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(x-\mu_k)^T \Sigma^{-1}(x-\mu_k)\right]}{\sum_{j=1}^{K} \frac{1}{2\pi^{d/2}|\Sigma|^{1/2}} \cdot \exp\left[-\frac{1}{2}(x-\mu_j)^T \Sigma^{-1}(x-\mu_j)\right]}$$

since $\Sigma_k^{-1} = \Sigma$ as constants

$$\gamma_k = \left\{ \sum_{j=1}^{K} (\pi_j/\pi_k) \cdot \exp\left[-\frac{1}{2}(x-\mu_j)^T \cdot \Sigma^{-1} \cdot (x-\mu_j) + \frac{1}{2}(x-\mu_k)^T \cdot \Sigma^{-1} (x-\mu_k)\right] \right\}^{-1}$$

M-step:

$$\frac{d \ln p(x|\pi, \mu, \Sigma^{-1})}{d\mu_k} = \sum_{n=1}^{N} \underbrace{\frac{\pi_k \mathcal{N}(x|\mu_k, \Sigma_k^{-1})}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x|\mu_j, \Sigma)}}_{\gamma_k} \cdot \Sigma^{-1}(x-\mu_k) = 0 \quad (*)$$

① $\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_k x^{(n)}$ (**)

(*) $\Rightarrow$ $0 = \sum_{n=1}^{N} \gamma_k (x-\mu_k)$

$\sum_{n=1}^{N} \gamma_k \mu_k = \sum_{n=1}^{N} \gamma_k x^{(n)}$ (**)

(**) $\Rightarrow$ $\mu_k \cdot \sum_{n=1}^{N} \gamma_k = \mu_k \cdot N_k$

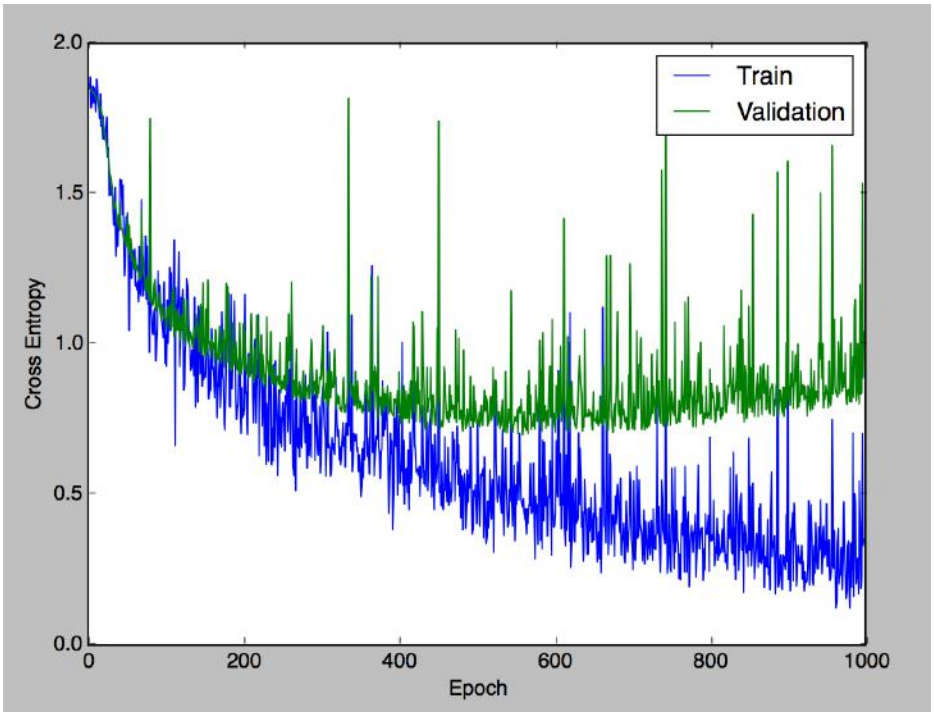
② $N_k = \sum_{n=1}^{N} \gamma_k$.

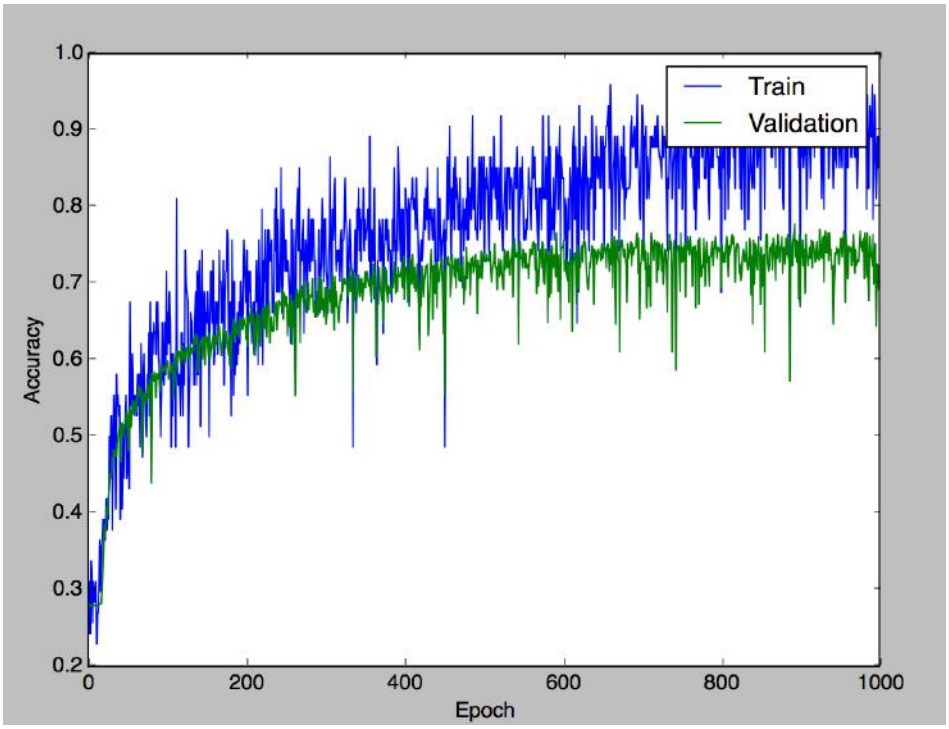
③ $\Sigma_k = \Sigma$

④ $\pi_k = N_k / N$

Q3.1

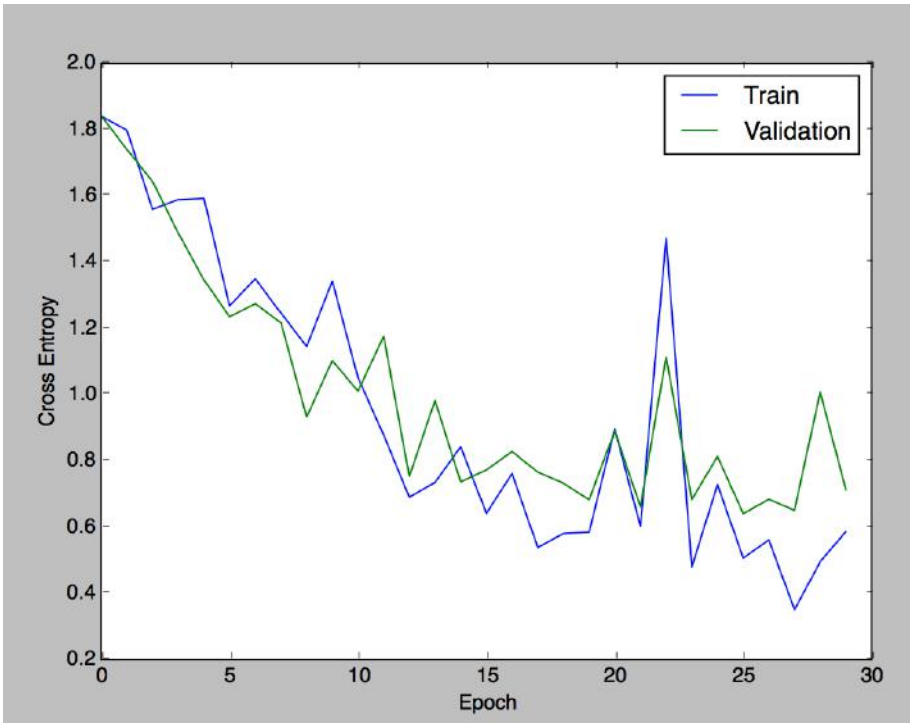
Plot of NN

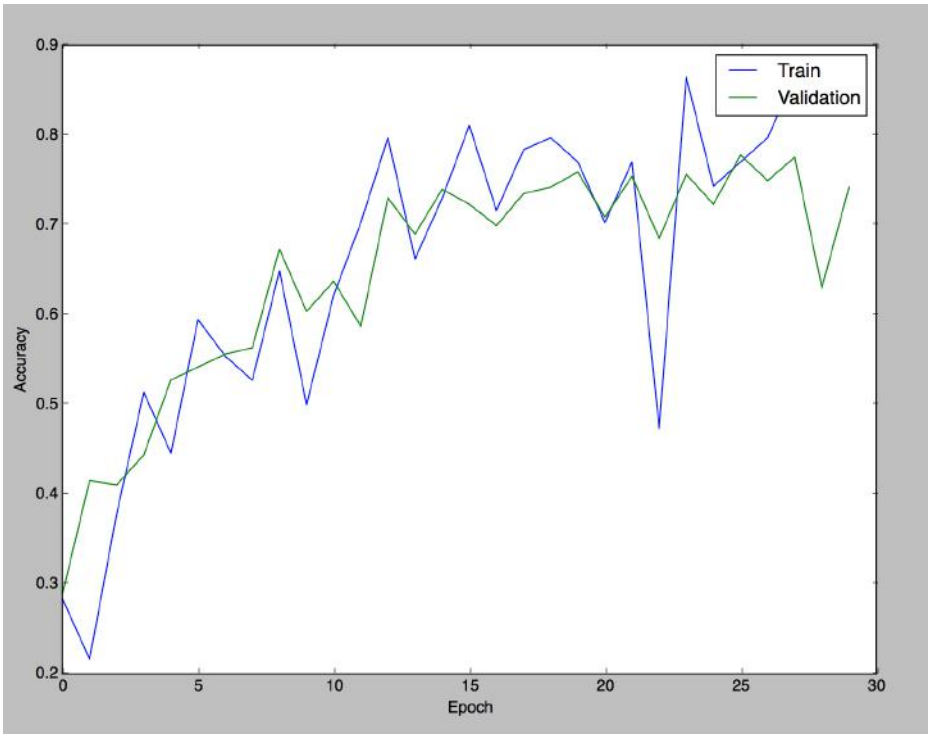




CE: Train 0.45940 Validation 1.04236 Test 1.05741
Acc: Train 0.82573 Validation 0.69212 Test 0.68831

Plot of CNN





CE: Train 0.46713 Validation 0.71247 Test 0.66317
Acc: Train 0.82662 Validation 0.74224 Test 0.75325

CNN has much better results compare to regular all connect nn

Q3.2

For NN:

Learning rate	Validation ACC	Validation CE
0.001	0.580	1.127
0.01	0.716	1.071
0.1	0.6086	1.396
0.5	0.338	1.859
1	0.338	1.841

Momentum	Validation ACC	Validation CE
0.0	0.716	1.071
0.5	0.687	0.933
0.9	0.730	1.851

Mini-batch size	Validation ACC	Validation CE
1	0.728	2.13
10	0.730	2.63
100	0.716	1.071
500	0.661	0.860
1000	0.587	1.091

Best parameters:
Learning rate:0.01
Momentum: 0.9
Mini-batch: 10

For CNN:

Learning rate	Validation ACC	Validation CE
0.001	0.279	1.824
0.01	0.589	1.147
0.1	0.771	0.685
0.5	0.279	1.876
1	0.279	1.872

Momentum	Validation ACC	Validation CE
0.0	0.792	0.639
0.5	0.783	0.696
0.9	0.280	1.858

Mini-batch size	Validation ACC	Validation CE
1	0.105	Nan
10	0.105	nan
100	0.7923	0.639
500	0.635	1.082
1000	0.508	1.449

Best parameters:
Learning rate:0.01
Momentum: 0
Mini-batch: 100

CNN prefer larger batch size and smaller momentum compare with NN.
CNN get a better Cross Entropy compare with NN and also, the lowest CE is always appear with the largest accuracy which is more reasonable.

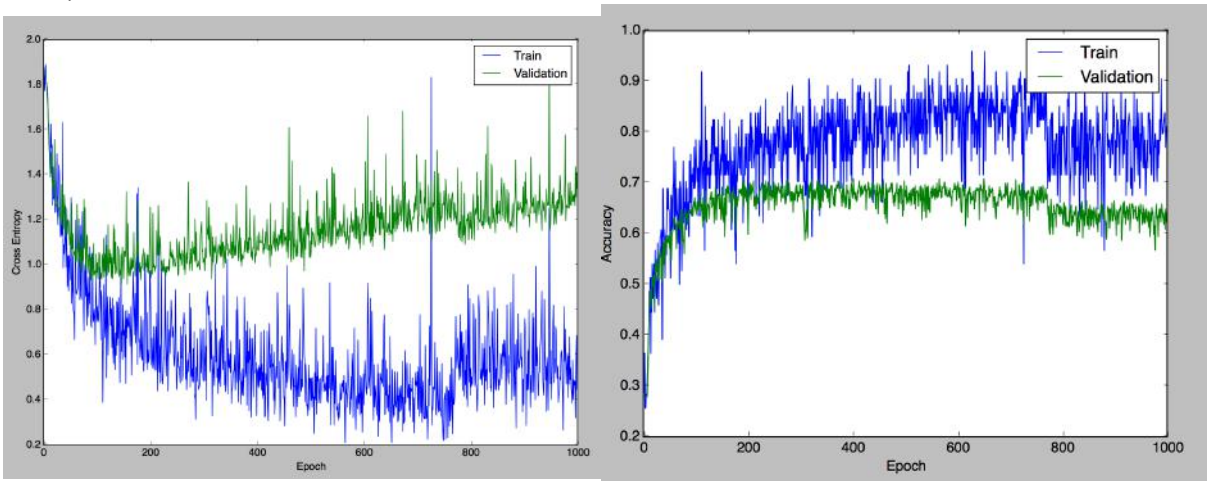
Q3.3

For NN

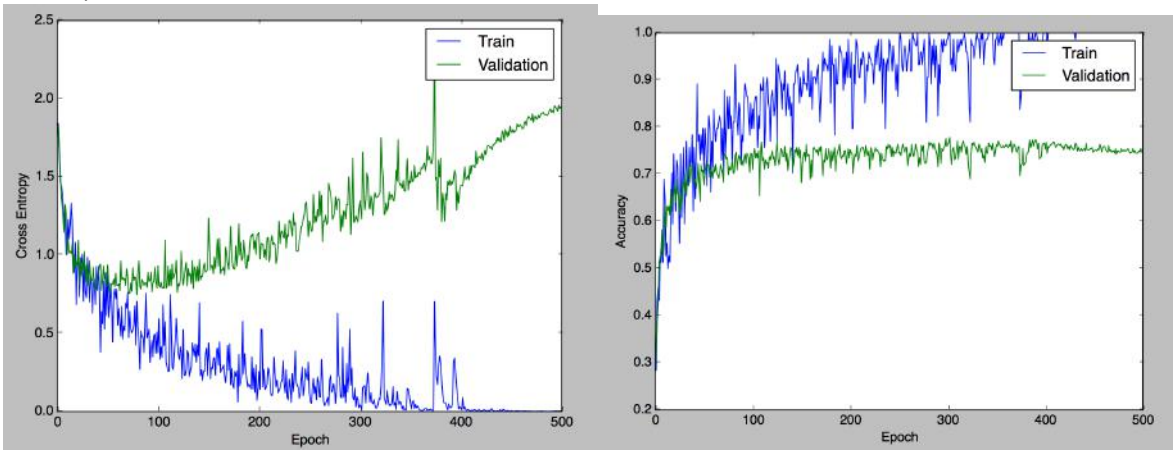
Hidden Layer	Validation ACC	Validation CE
8,16	0.623	1.340
32,64	0.747	1.949
48,96	0.735	1.467

As hidden units increase the results is better, and reach to maximum at 32 and 64 hidden units, however the cross entropy increased which indicate that the margin between each classification is relatively small compare to less hidden units.

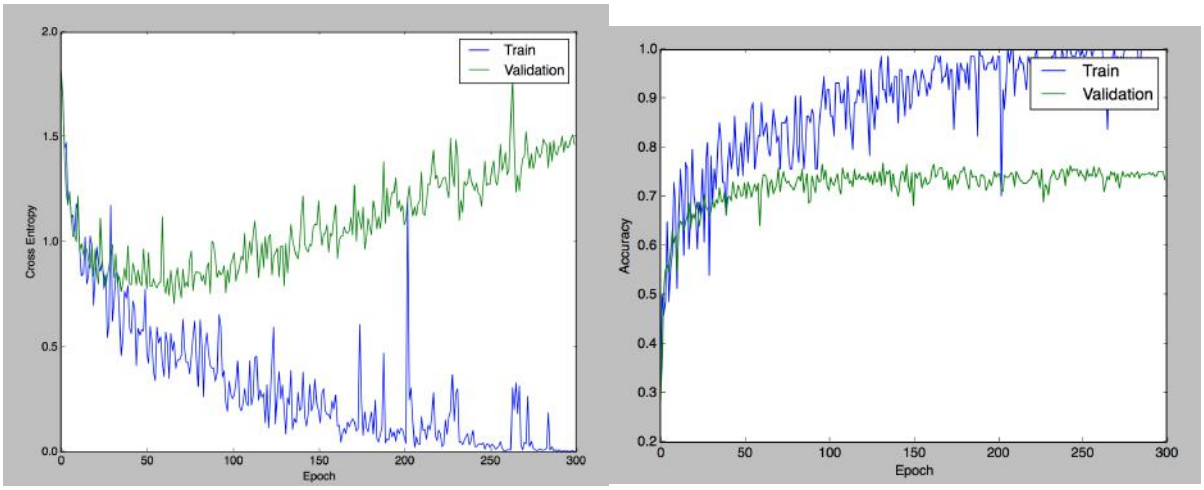
NN 8,16



NN 32,64



NN 48,96



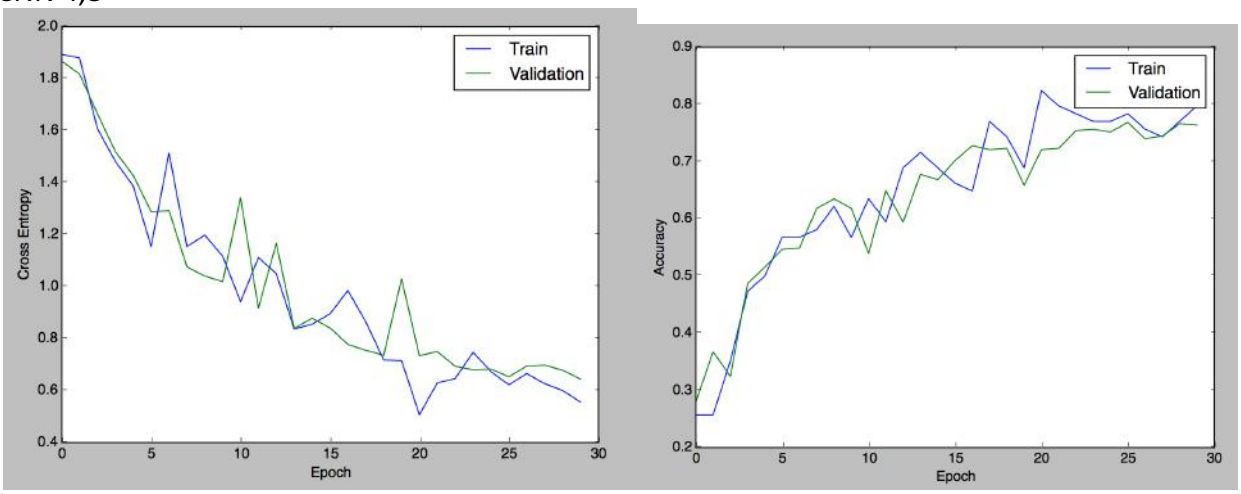
In all connected NN, the convergence is more quickly in higher number of hidden units, as units decrease, it needs more epoch to converge.

For CNN

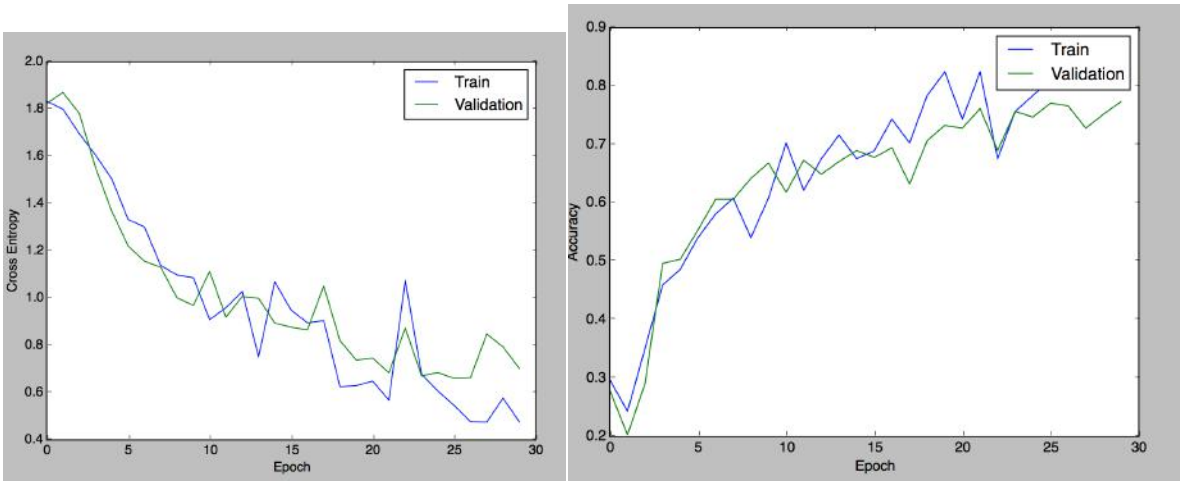
Filter	Validation ACC	Validation CE
4,8	0.764	0.644
16,16	0.773	0.701
16,32	0.735	1.467

As number of filter increase the performance is increasing every little, also the Cross entropy is increasing as filters increase. But the effect is much better than all connect neural network.

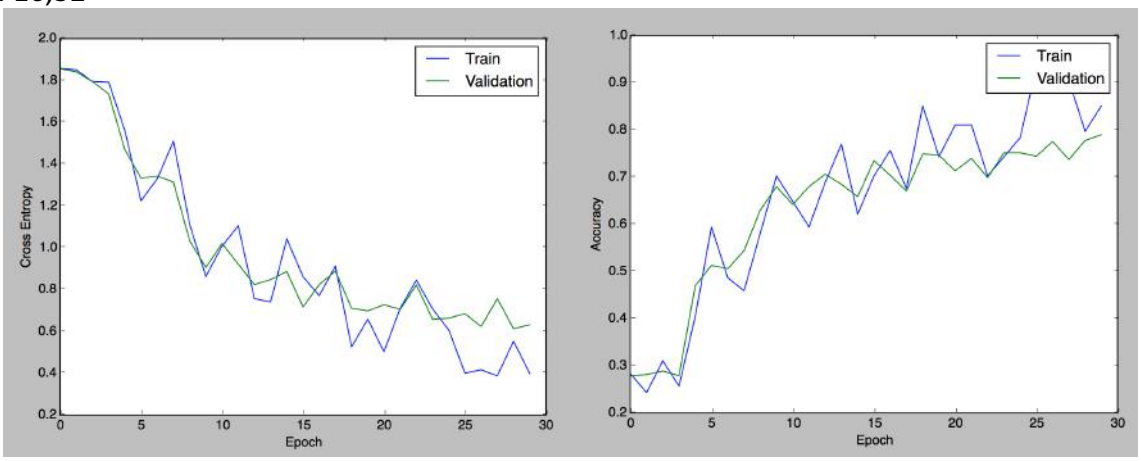
CNN 4,8



CNN 16,16



CNN 16,32



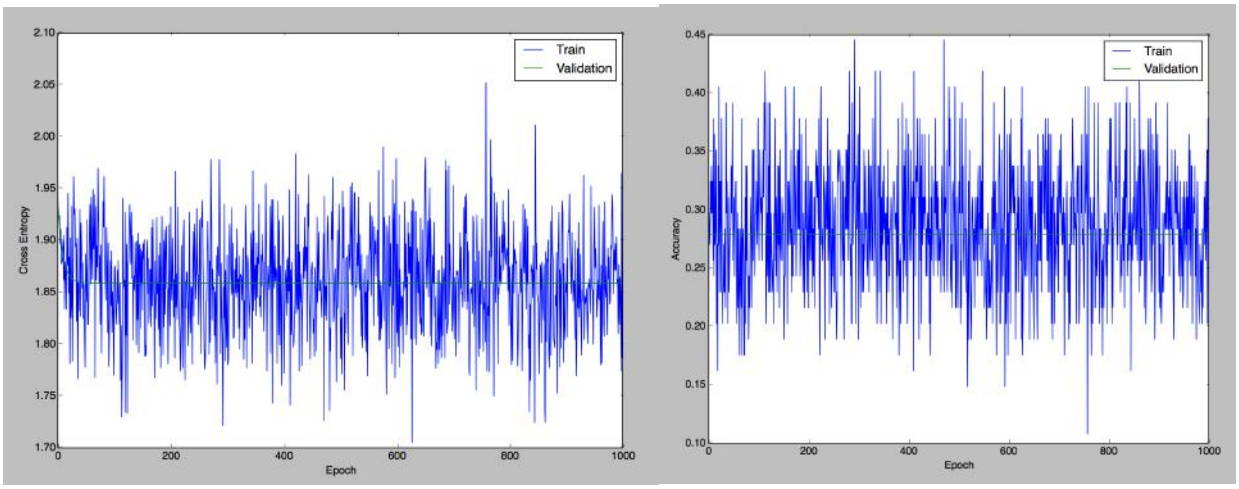
The convergence is not really effected by the increment of filters; it might be there is no useful features to distinguish among different classes. However, CNN take much more less epoch than NN to reach convergence point. But for each epoch CNN takes longer time compare with NN.

Q3.4

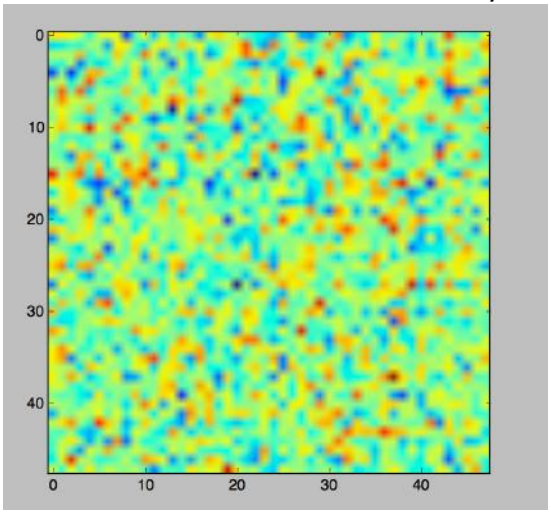
Choosing Hidden unit of 1,1
# of para in NN: 48*48 * 1 + 1 + 1*1+1 = 2307

Choosing Filter of 45,45, Filter size = 5*5
# of para in CNN: 2* (5*5*45 + 45) = 2340

NN:

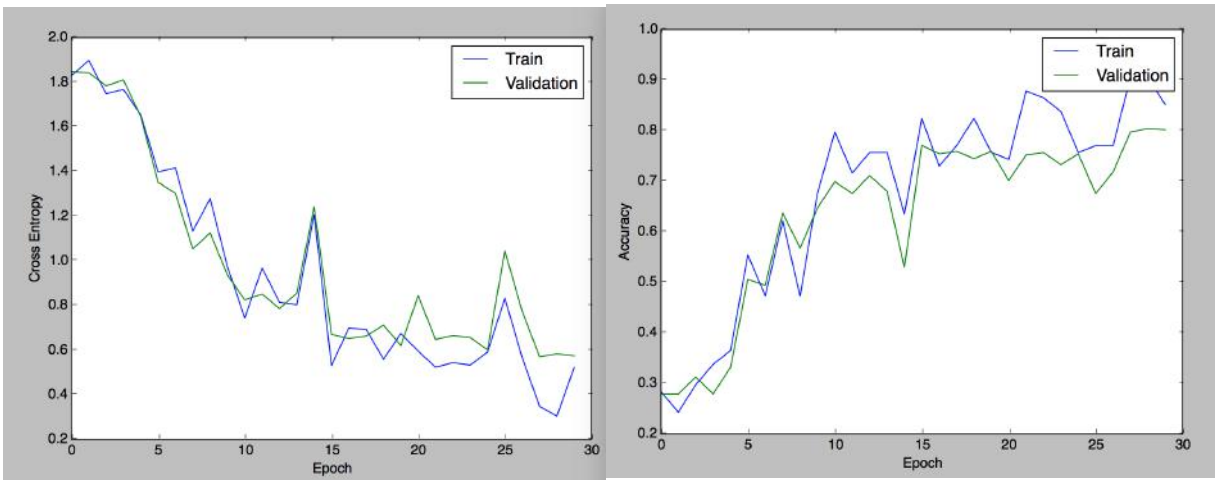


As you can tell from above there is no improvement after 1000 epoch, with 2307 Weights the all connect NN performance is really poor.
Only with much lager number of weights it provides good results. E.g. 48*48*16 + 16 + 16*16*8+8 = 38936 with 0.6 accuracy

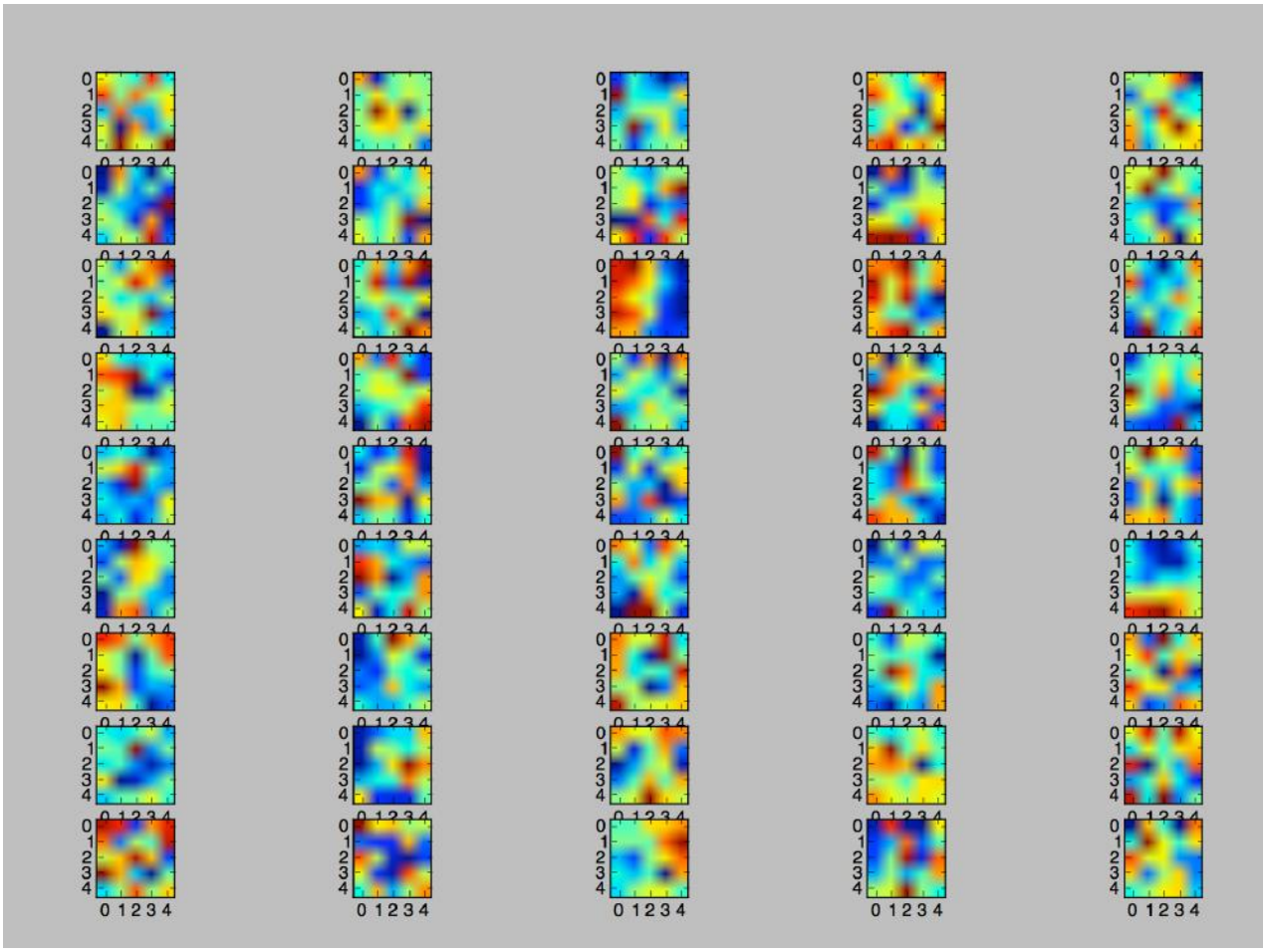


This is the first layer pic which generated by 48*48 weights. It is more like random generated picture without any pattern.

CNN:



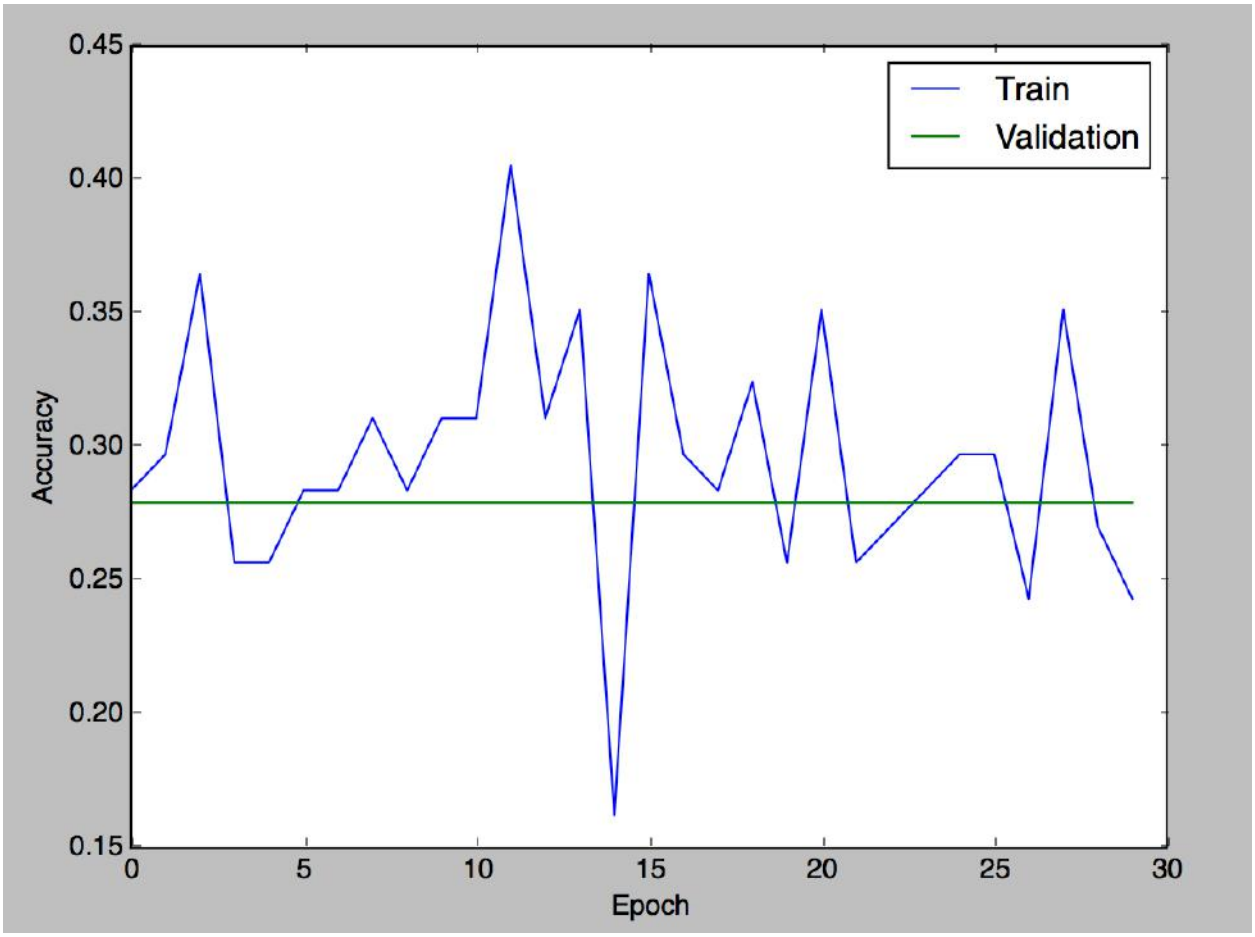
Under the same parameter, the CNN performance is much better than all connect NN. CNN is definitely leading to better generalization.



This is first 45 layer CNN filters. These images have more pattern compared with NN previously.

Q3.5

CNN with eps = 0.001



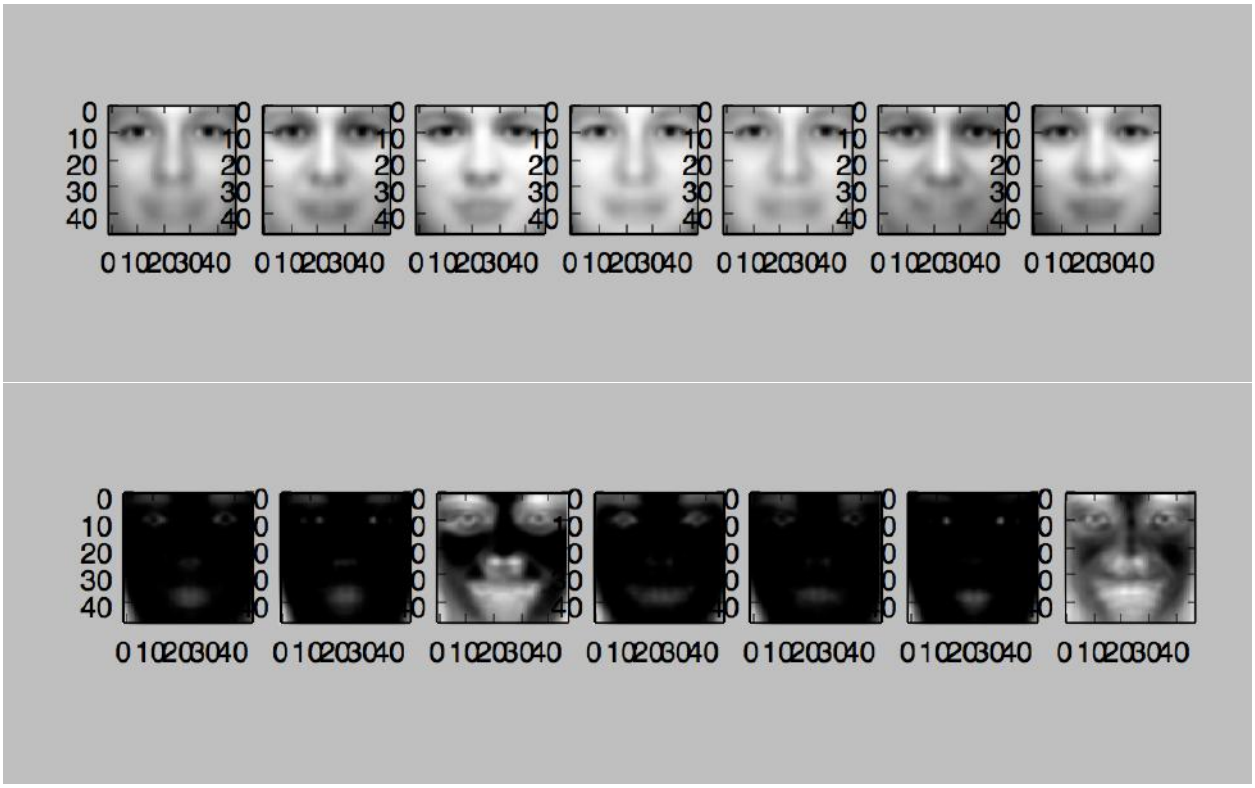
The accuracy is don't have any change in 30 epoch.

The classification would be wrong if the number is just slightly large than 1/7. Cause there is 1/7 probability to be right even with guess.

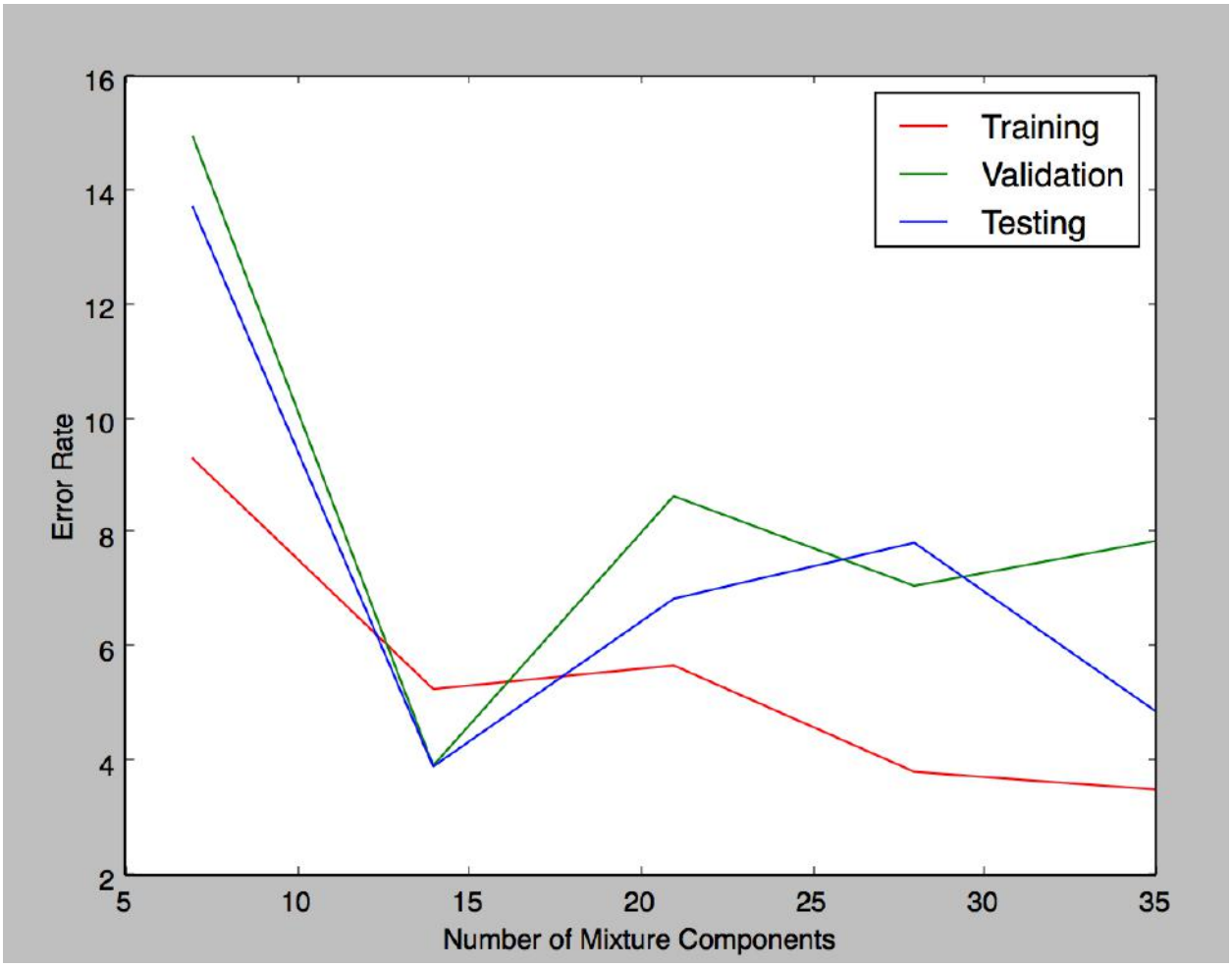
Q4

4.2

4.3



4.4



As the number of clusters increase, there are more hidden variables increase the accuracy of the output.
The test set is generally have decreasing trends as components increase.