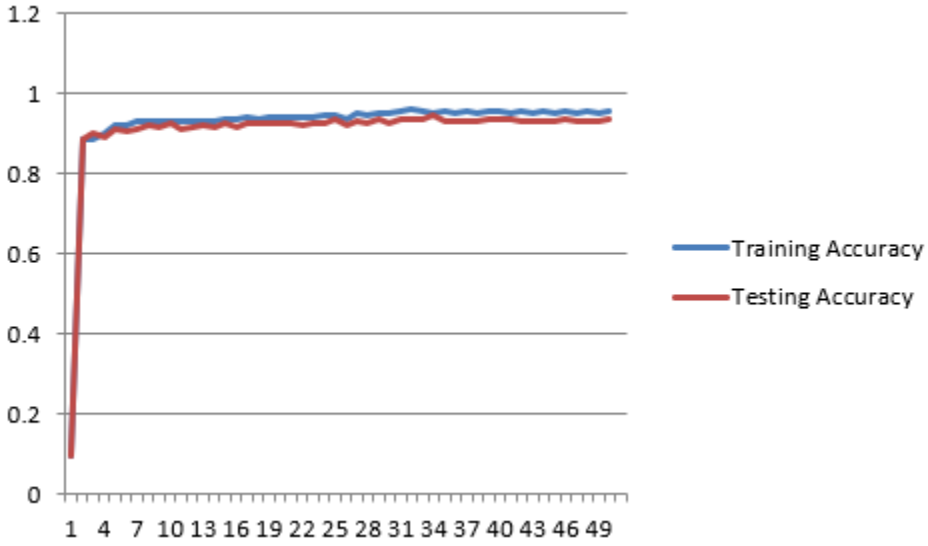
Experiment 1

n=20:

Graph:



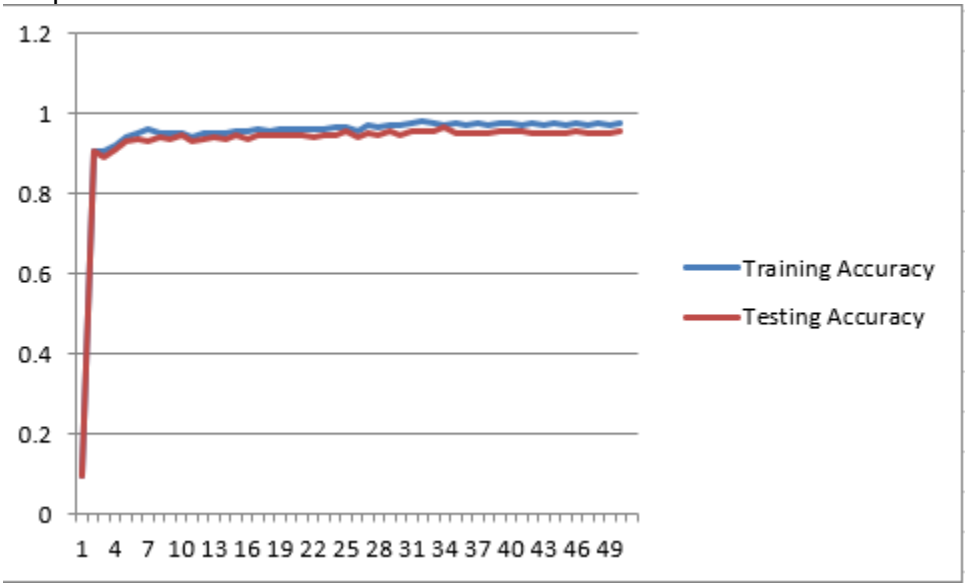
Confusion Matrix:

```
Confusion Matrix :

[[ 949    0    6    2    0    2    9    2    9    1]
 [   0 1110    4    4    1    2    1    1   12    0]
 [  15    2  923   26   13    9    3   19   21    1]
 [   3    0   13  922    2   37    2    7   16    8]
 [   1    0    4    0  935    1   13    2    7   19]
 [  12    2    0   22    0  815   10    6   19    6]
 [  11    3    7    1    5    7  914    0   10    0]
 [   1    6   20    9    7    0    0  965    6   14]
 [   6    2    6   11    5   19   11    6  904    4]
 [   9    7    2   13   36   10    2   19   15  896]]
```

n=50:

Graph:



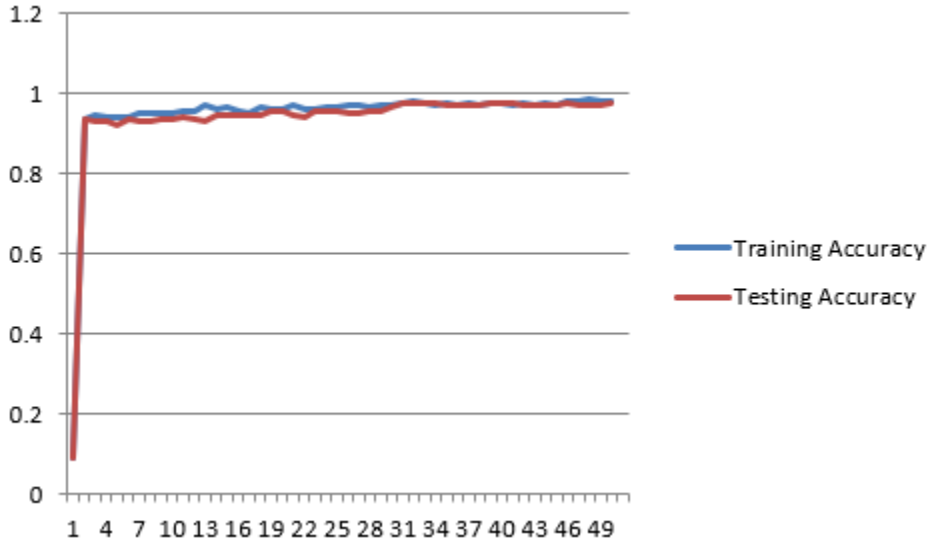
Confusion Matrix:

```
Confusion Matrix :

[[ 964    1    2    2    1    1    2    3    2    2]
 [   0 1115    4    4    0    1    2    2    7    0]
 [   7    3  995   10    3    0    2    5    6    1]
 [   0    0   12  966    1    6    1    8   14    2]
 [   1    1    6    1  935    0    8    1    7   22]
 [   9    0    2   18    0  838    9    4    9    3]
 [   9    3    4    1    3   10  922    0    6    0]
 [   1    5   19    7    2    0    0  979    4   11]
 [   5    3    9   11    4    8    7    2  921    4]
 [   4    8    0   13   14    6    1    8   17  938]]
```

for n=100:

Graph:



Confusion Matrix:

```
Confusion Matrix :

[[ 967    1    2    1    1    2    3    1    1    1]
 [   0 1116    3    7    0    1    2    1    5    0]
 [   4    3 1000    7    2    1    2    8    5    0]
 [   0    0   10  980    1    6    0    4    7    2]
 [   1    0    4    0  954    0    6    1    2   14]
 [   6    0    0   13    3  852   10    2    5    1]
 [   7    2    2    1    7    3  932    0    4    0]
 [   0    4   13    2    4    0    0  995    0   10]
 [   6    1    6    9    4    6    3    4  933    2]
 [   6    4    4    5   15    6    0    6    5  958]]
```

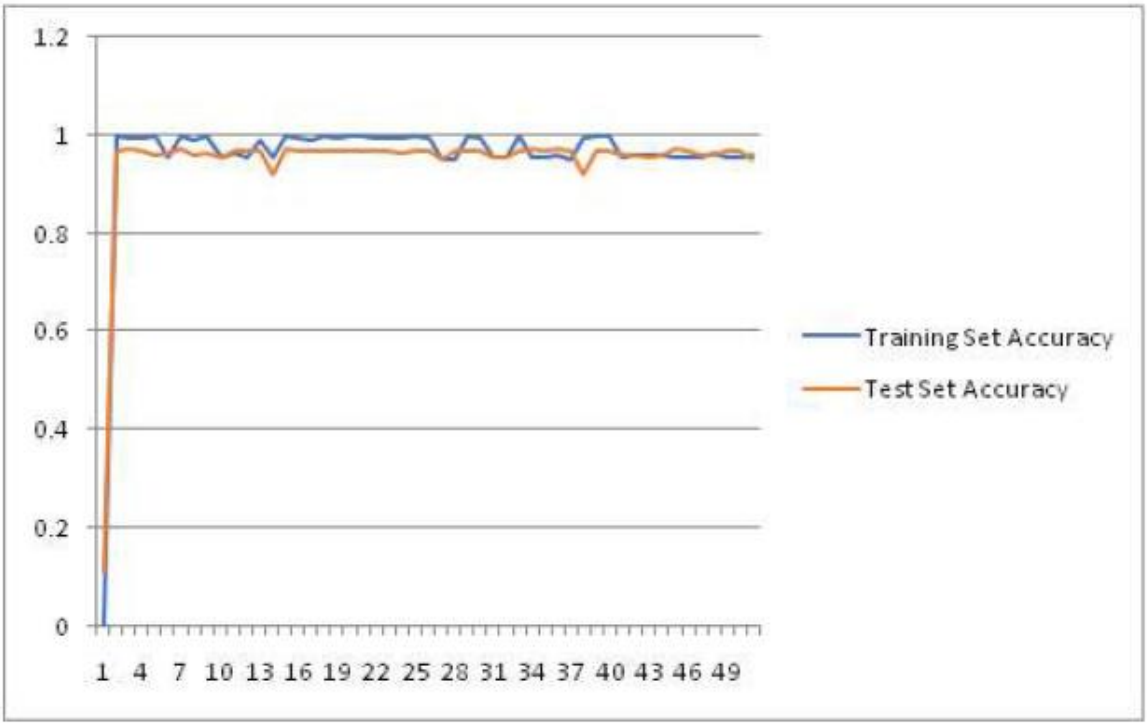
(1) How does the number of hidden units affect the final accuracy on the test data?
The accuracy increases as we increase the number of hidden layer.
(2) How does it affect the number of epochs needed for training to converge?
As we increase the number of layers the epochs taken to converge the data are less. So, it converges in less number of epochs.
(3) Is there evidence that any of your networks has over-fit to the training data? If so, what is that evidence?

Yes, there is over fitting for first few epochs and it can be seen from graph also that training accuracy is more than test accuracy.

(4) How do your results compare to the results obtained by your perceptron in HW 1?

The accuracy has increased, it also converges early.

Experiment 2-
momentum=0
Graph:



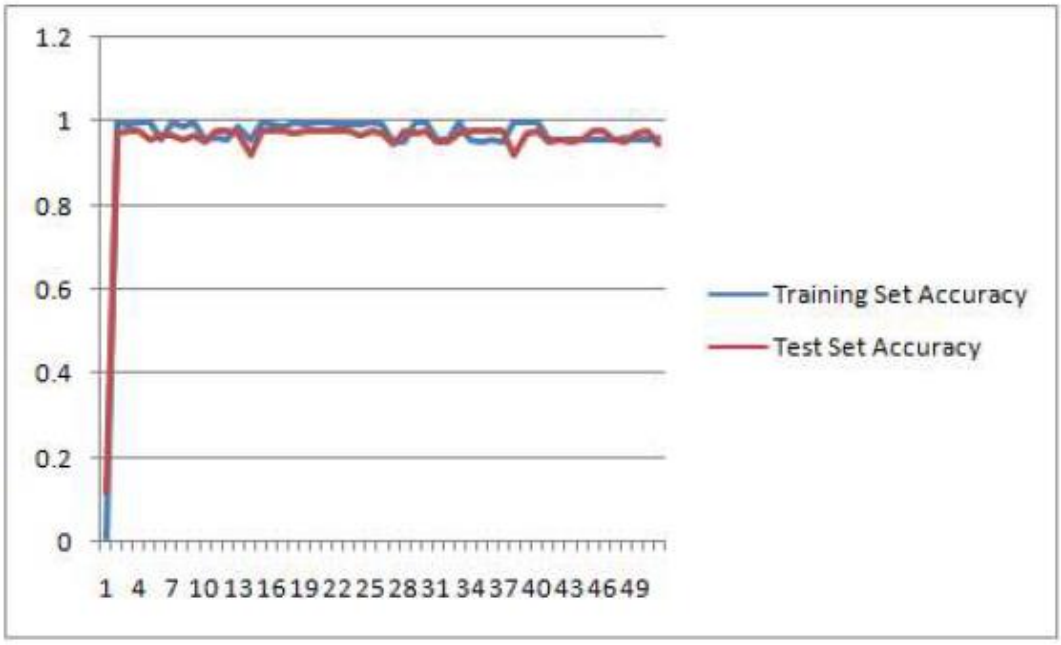
Confusion Matrix:

```
Confusion Matrix :

[[ 964    0    0    2    0    4    3    2    2    3]
 [   0 1123    4    1    0    1    3    1    2    0]
 [   4    3 1000    4    3    1    2    7    8    0]
 [   0    1    6  989    1    3    1    4    4    1]
 [   1    0    2    1  954    0    7    2    2   13]
 [   3    1    0   15    1  857    6    1    5    3]
 [   7    3    1    1    3    4  934    1    4    0]
 [   2    8   11    7    3    1    0  988    1    7]
 [   3    1    6    8    6    8    2    3  934    3]
 [   4    5    1   10   14    4    1    6    3  961]]
```

momentum=.25

Graph:

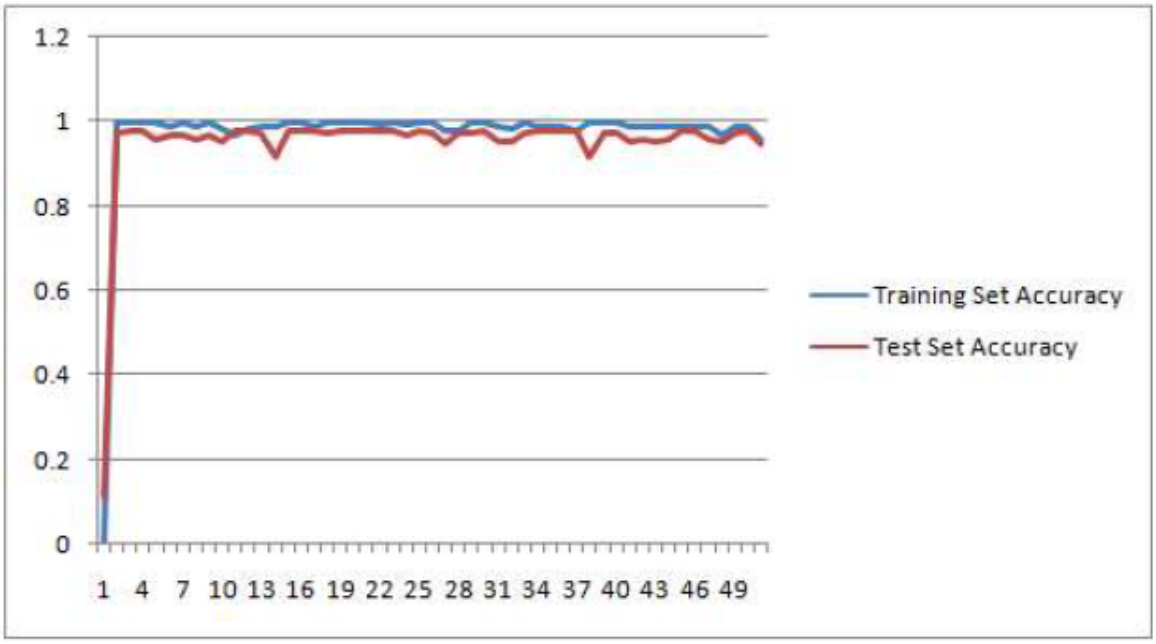


Confusion Matrix:

```
Confusion Matrix :

[[ 963    0    1    2    1    1    6    2    2    2]
 [   0 1125    3    1    0    1    3    0    2    0]
 [   4    2  998    3    4    1    2    9    8    1]
 [   0    1    4  990    1    2    0    3    5    4]
 [   1    0    3    1  949    0    6    2    2   18]
 [   2    1    1   14    3  855    8    1    4    3]
 [   6    3    0    1    4    5  934    0    5    0]
 [   0   10   11    4    3    1    1  985    2   11]
 [   3    1    4    7    4    7    5    4  938    1]
 [   5    7    1    9   17    7    1    7    3  952]]
```

momentum=.5
Graph:



Confusion Matrix:

```
[[ 966    0    0    1    0    4    5    1    2    1]
 [   0 1118    2    2    0    1    5    1    6    0]
 [   4    2 1007    3    3    0    1    7    5    0]
 [   0    0    9  984    0    3    0    5    7    2]
 [   1    1    6    1  953    0    3    2    1   14]
 [   2    1    0   14    3  855    7    2    4    4]
 [   6    2    2    1    2    7  932    0    6    0]
 [   2    8   15    3    0    0    0  990    1    9]
 [   4    1    5    6    6    4    4    1  941    2]
 [   3    4    2   10   12    5    1    8    4  960]]
```

(1) How does the momentum value affect the final accuracy on the test data?
When there was no momentum the accuracy achieved was maximum and for the other values the accuracy was the similar at the convergence point.

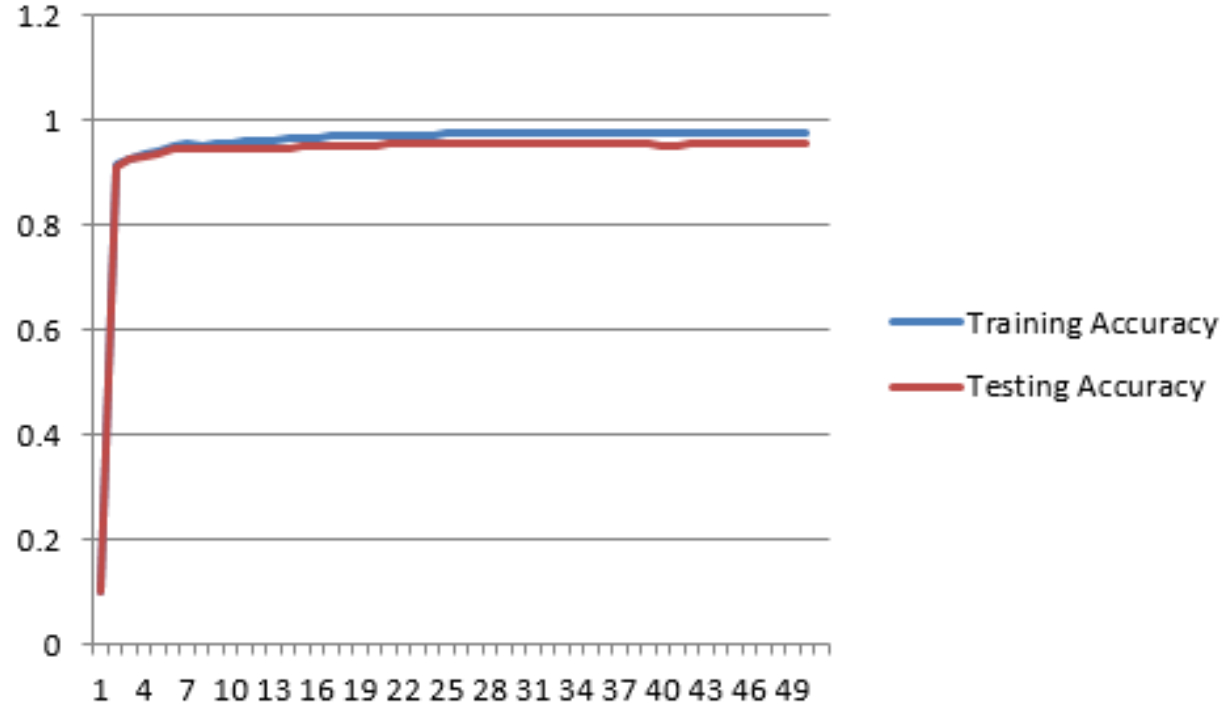
(2) How does it affect the number of epochs needed for training to converge?
The network which was using the higher value of the momentum converged earlier.
(3) Again, is there evidence that any of your networks has over fit to the training data? If so, what is that evidence?

We can say that there was overfitting as for some instances the testing accuracies was less than that of the training accuracies. It can be seen from the graph as well.

Experiment 3

Half

Graph



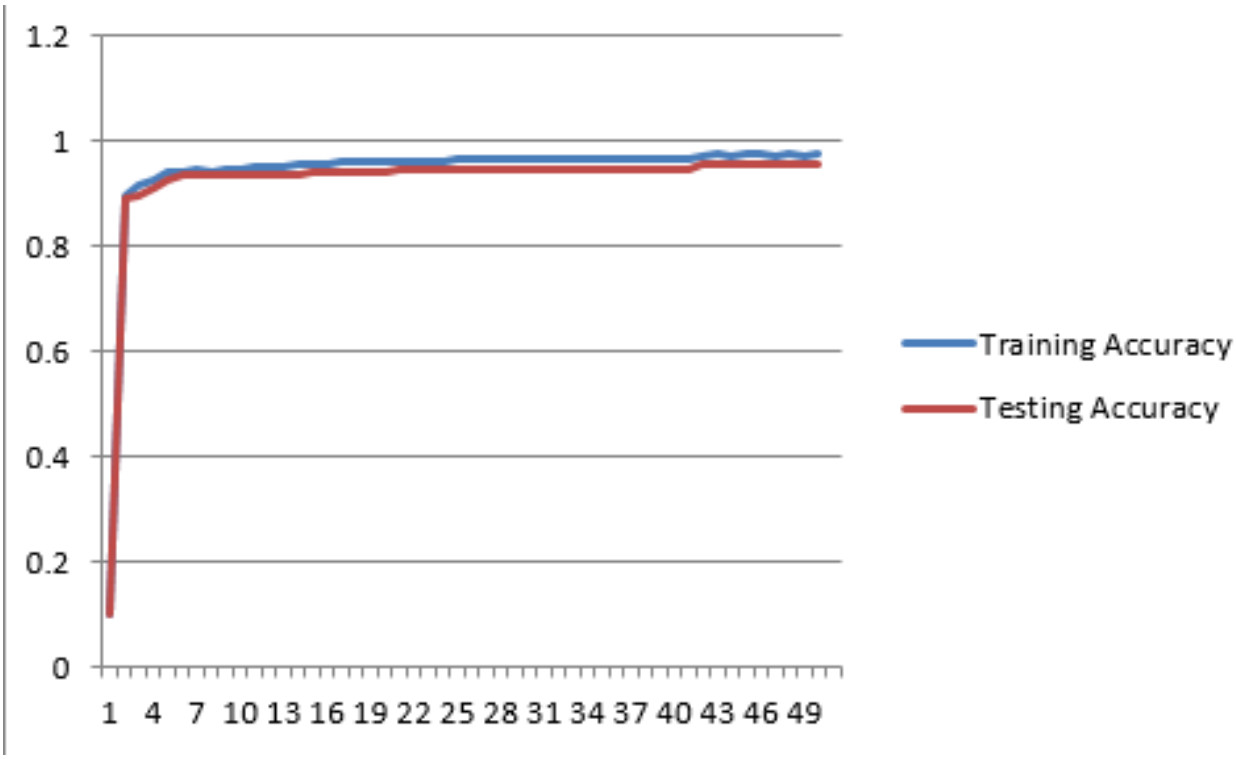
Confusion Matrix

```
Confusion Matrix :

[[ 963    0    2    4    0    4    4    1    2    0]
 [   0 1118    4    3    0    2    3    1    4    0]
 [   8    2  997    8    3    0    1    9    4    0]
 [   0    1   10  970    1    9    0    8    5    6]
 [   1    1    3    0  948    0    8    2    2   17]
 [   5    0    1   11    0  855    4    3    9    4]
 [  11    4    2    2    2    9  920    2    6    0]
 [   3    9   12    1    2    0    2  984    3   12]
 [   5    0    5   15    4   13    4    6  919    3]
 [   4    7    1    7   22    4    2    7    6  949]]
```

Quarter

Graph



Confusion Matrix

```
Confusion Matrix :

[[ 961    0    2    2    0    3    5    3    3    1]
 [   0 1118    5    2    0    1    4    1    4    0]
 [  12    0  976   10    9    0    8    9    7    1]
 [   2    0   10  963    1   13    0   10    9    2]
 [   2    0    2    0  940    2    7    0    5   24]
 [   5    1    0   24    3  824   12    5   11    7]
 [  14    3    2    2    7    7  911    1   11    0]
 [   2    8   17    3    6    1    0  979    1   11]
 [   5    1    4   17    4   10    6    8  914    5]
 [   5    9    3   11   26    7    1   13    5  929]]
```

(1) How does the size of the training data affect the final accuracy on the test data?
The Final accuracies for quarter and half were almost same for initial runs and there was little difference, accuracy was higher for half then quarter.
(2) How does it affect the number of epochs needed for training to converge?
The epochs for the convergence of quarter was lower and for half it was high. So, quarter converged earlier.
(3) Again, is there evidence that any of your networks has over fit to the training data? If so, what is that evidence?
The training accuracy was more than test for initial runs, so we can say that yes there was overfitting.