Name: Anurag Panwar (U4641-6226)

1. (50 pts) Implement a convolutional neural network that has one convolutional layer, one max pooling layer and two fully-connected layers. The convolutional layer has 10 filters (or kernels with size of 5x5) and their strides are 2. The pooling is 4x4 pooling. The two full-connected layers have 100 and 10 neurons, respectively.
a. (10 pts) Report the number of parameters you need to train.
b. (30 pts) Train the network with the MNIST dataset
c. (10 pts) Visualize the 10 filters.

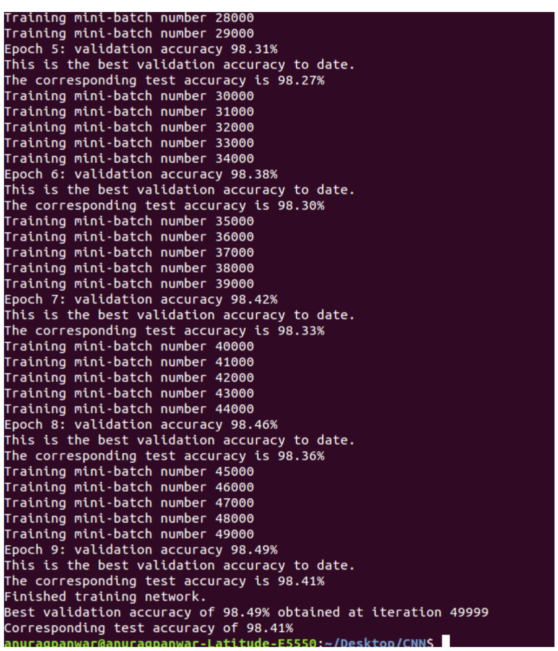
a) For each feature map(filter/kernel) we need $25=5\times5$ shared weights, plus a single shared bias. So each feature map requires 26 parameters. If we have 10 feature maps that's a total of $10\times26=260$ parameters defining the convolutional layer.

FullyConnectedLayer(n_in=10*3*3, n_out=100)
For first fully-connected layer, we have 10*3*3 input neurons and 100 output neurons. Total number of weights = 90*100 = 9,000 and it has bias for each neurons which is equal to 100.
Total parameters = 9,100

SoftmaxLayer(n_in=100, n_out=10)], mini_batch_size)
For second fully connected layer, we have 100 as input neurons and 10 as output neurons.
Total number of neurons = 100*10 = 1,000 and it has bias for each neurons which is equal to 10.
Total parameters = 1,010

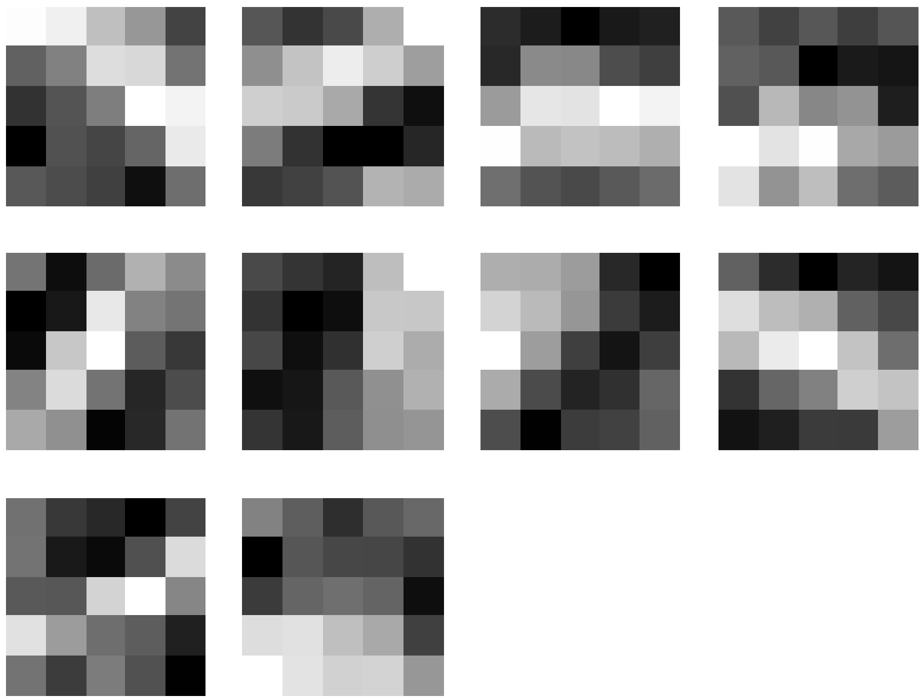
The aggregated total number of parameters are 260+9,100+1,010 = 10,370

b) For mini batch size of 10
number of epoch = 10 (5000 iterations each)
Best validation accuracy of 98.49% obtained at iteration 49999
Corresponding test accuracy of 98.41%



c)Final receptive field (kernel/filter)

Total number of filters = 10



2. (50 pts) Implement another convolutional neural network that has two convolutional layers, two max pooling layers, and two fully-connected layers.
a. (20 pts) Report the design of your CNN (number of filters, size of the filters, stride, padding, pooling, and number of neurons in each layer)
b. (10 pts) Report the number of parameters you need to train.
c. (10 pts) Train the network with the MNIST dataset
d. (10 pts) Visualize the filters for the first convolutional layer.

a) Design of the CNN is as follow:
First CNN layer with max-pooling
Number of filters for 1^{st} convolutional layer = 20

Size of filter for 1st convolutional layer = 5*5
Stride = 1
Padding = 0
Max-pooling = (2,2)

We can compute the spatial size of the output volume as a function of the input volume size (W), the receptive field size of the Conv Layer neurons (F), the stride with which they are applied (S), and the amount of zero padding used (P) on the border. You can convince yourself that the correct formula for calculating how many neurons "fit" is given by $(W-F+2P)/S+1$.

Number of input neurons = 28*28 = 784
Number of neurons in 1st Convolution layer = (((28-5+2*0)/1+1)*((28-5+2*0)/1+1))
=24*24*20 = 11520
Number of neurons after 1st max pooling = 12*12*20 = 2880

Second CNN layer with max-pooling
Number of filters for 1st convolutional layer = 40
Size of filter for 1st convolutional layer = 5*5
Stride = 1
Padding = 0
Max-pooling = (2,2)

Image shape for 2nd convolution layer = 20*12*12 = 2880
Number of neurons in 2nd Convolution layer = (((12-5+2*0)/1+1)*((12-5+2*0)/1+1))
=8*8*40 = 2560
Number of neurons after 2nd max pooling = 4*4*40 = 640

1st **FullyConnectedLayer**
Number of neurons = 100

2nd FullyConnectedLayer
Number of neurons = 10

```
    net = Network([
  ConvPoolLayer(image_shape=(mini_batch_size, 1, 28, 28),
          filter_shape=(20, 1, 5, 5),
          poolsize=(2, 2),
          activation_fn=ReLU),
ConvPoolLayer(image_shape=(mini_batch_size, 20, 12, 12),
          filter_shape=(40, 20, 5, 5),
          poolsize=(2, 2),
          activation_fn=ReLU),
FullyConnectedLayer(n_in=40*4*4, n_out=100, activation_fn=ReLU),
SoftmaxLayer(n_in=100, n_out=10)], mini_batch_size)
```

b) For each feature map(filter/kernel) we need $25=5\times5$ shared weights, plus a single shared bias. So each feature map requires 26 parameters. If we have 20 feature maps that's a total of $20\times26=520$ parameters defining the convolutional layer.

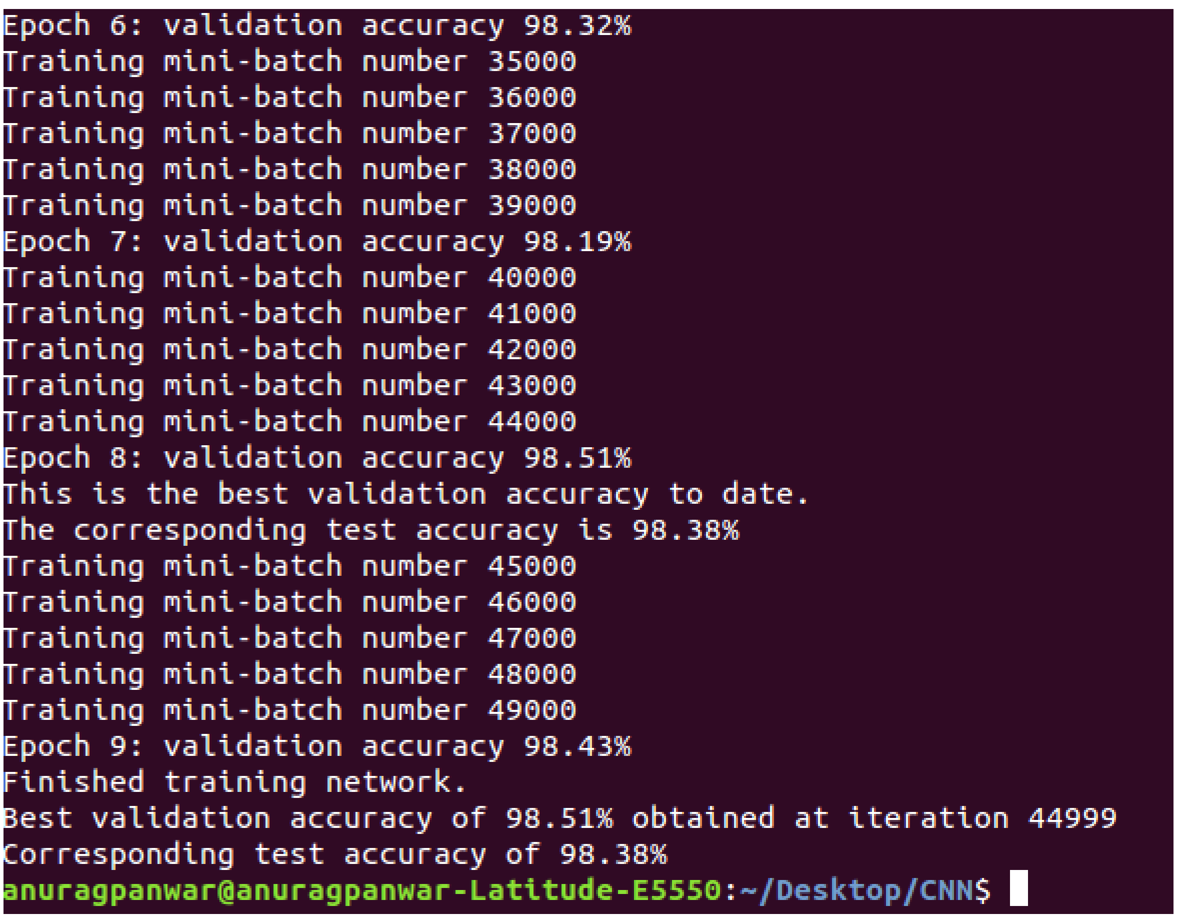
For 2nd convolution layer, we need $25=5\times5*20$ shared weights, plus a single shared bias. So each feature map requires 501 parameters. If we have 40 feature maps that's a total of $40\times501= 2040$ parameters defining the convolutional layer.

FullyConnectedLayer(n_in=40*4*4, n_out=100, activation_fn=ReLU),
For first fully-connected layer, we have 40*4*4 input neurons and 100 output neurons. Total number of weights = 640*100 = 64,000 and it has bias for each neurons which is equal to 100. Total parameter = 64,100

SoftmaxLayer(n_in=100, n_out=10)], mini_batch_size)
For second fully connected layer, we have 100 as input neurons and 10 as output neurons. Total number of neurons = 100*10 = 1,000 and it has bias for each neurons which is equal to 10. Total parameter = 1,010

The aggregated total number of neurons are 520+2,040+64,100+1,010 = 67,670

c) For mini batch size of 10
number of epoch = 9 (5000 iterations each)
Best validation accuracy of 98.51% obtained at iteration 44999
Corresponding test accuracy of 98.38%



```
Epoch 6: validation accuracy 98.32%
Training mini-batch number 35000
Training mini-batch number 36000
Training mini-batch number 37000
Training mini-batch number 38000
Training mini-batch number 39000
Epoch 7: validation accuracy 98.19%
Training mini-batch number 40000
Training mini-batch number 41000
Training mini-batch number 42000
Training mini-batch number 43000
Training mini-batch number 44000
Epoch 8: validation accuracy 98.51%
This is the best validation accuracy to date.
The corresponding test accuracy is 98.38%
Training mini-batch number 45000
Training mini-batch number 46000
Training mini-batch number 47000
Training mini-batch number 48000
Training mini-batch number 49000
Epoch 9: validation accuracy 98.43%
Finished training network.
Best validation accuracy of 98.51% obtained at iteration 44999
Corresponding test accuracy of 98.38%
anuragpanwar@anuragpanwar-Latitude-E5550:~/Desktop/CNN$
```

d) Final receptive field (kernel/filter)

Total number of filters = 20