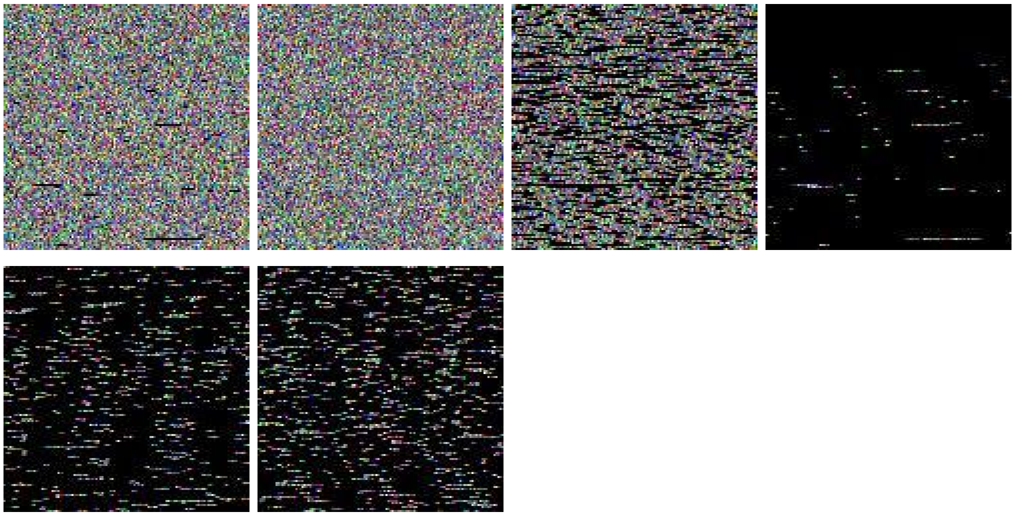
SMAI ASSIGNMENT 2

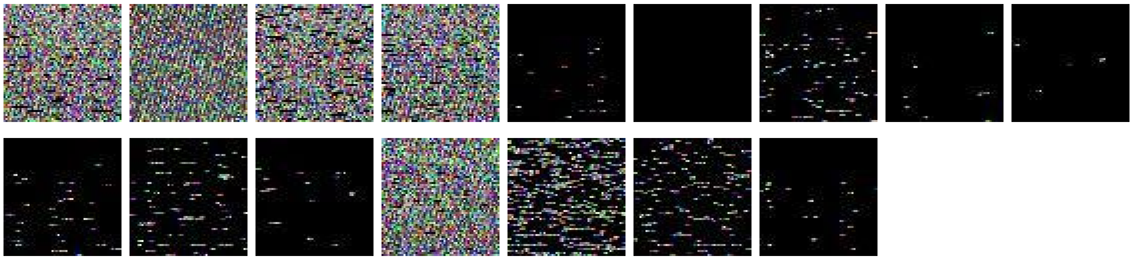
- Priyansh Agrawal
201502168

1.1 Images after convolutions are smaller because we do not add padding. After first convolution images are faint outline of original image. We produce one long vector at last whose size is reduced using max pooling. Some sample images are below respectively after first convolution, first pooling, second convolution and second pooling.
Pooling layer downsamples the volume spatially, independently in each depth slice of the input volume.

INPUT IMAGE



6 LAYER AFTER CONV-RELU-POOL (using 6 filters)



16 LAYERS AFTER CONV-RELU-POOL(using 16 filters)

1.2

1. What are the number of parameters in convolution layers with K filters each of size 3*w*h.

Ans. Total number of parameters are 3whK + K
Where 3whK is for the weight and K is for the bias

2. What are the number of parameters in a max pooling operation?

Ans. Number of parameters is zero as no value needs to be stored
Pooling take the size of the pooling matrix. Some time it also takes stride as argument.
In our case stride is 2 and size is 2*2.

3. Which of the operations contain most number of parameters? (a) conv (b) pool (c) Fully connected layer (FC) (d) Relu
Option c)Fully Connected contains most number of parameters. As it we have to now take into account all the values as parameters rather than CONV layer in which small image is used.
Fully connected layer as it's size would be input size*number of hidden layer. When using cifar-10, 32*32*3 images compare small against number of parameters in Fully connected layer.

4. Which operation consume most amount of memory? (a) initial convolution layers (b) fully connected layers at the end

Ans. (a) Initial convolution layers because we have to store images too which consumes memory.

5.

With increase in learning rate accuracy decreases.

Batch Size = 50
Learning rate = 0.001

```
(‘Test score:’, 1.1326242254257202)
(‘Test accuracy:’, 0.6047)
```

Batch Size: 50
Learning rate = 0.01

```
('Test score:', 1.1468873929977417)
('Test accuracy:', 0.5977)
```

So accuracy decreases with increase in learning rate

6.

It is not smooth it is fluctuating for Batch size = 1
For batch size 50 it is smooth as shown in question before

Above plot is for batch size = 1

For batch size = 100

```
('Test score:', 1.236137379837036)
('Test accuracy:', 0.5618)
```

So if the value of batch size is too high or too low accuracy will decrease

7.

Accuracy generally increases but may behave erratically sometimes.
Number of convolution layer = 1

```
('Test score:', 1.4729675039291381)
('Test accuracy:', 0.4658)
```

So accuracy decreases with decrease in no of filters from 6 to 1

8.
It increases at first but after increasing it to very high number it starts overfitting data and accuracy decreases.
9.
Relu and tanh have really good accuracy compared to sigmoid.
Range of tanh is between -1 to 1 hence optimization over it is easier compared to sigmoid whose range is from 0 to 1. Relu being simple makes it easy to use but it should only be used in hidden layers because it do not model probability very well.

Activation Function	Accuracy

tanh	0.615
relu	0.636
sigmoid	0.324

10.

1. Rotating(like images and stuff).
2. Blurring images(blurring won’t change class).
3. Adding noise.
4. Applying filters of certain types.

2. 3-layer Neural Networks

1) Activation Functions

Two function used in this assignment in hidden layer are (Logistic) sigmoid and tanh. Both function lies between -1 and 1. Derivative of both functions is simple to implement. Both are non-linear hence help model non-linearity in data.

Sigmoid Activation function: It is a activation function of form **f(x) = 1 / 1 + exp(-x) .** Its Range is between 0 and 1. It is a S — shaped curve. It is easy to understand and apply but it has major reasons which have made it fall out of popularity -

- Vanishing gradient problem
- Secondly , its output isn’t zero centered. It makes the gradient updates go too far in different directions. **0 < output < 1, and it makes optimization harder.**
- Sigmoids saturate and kill gradients.
- Sigmoids have slow convergence.

Hyperbolic Tangent function- Tanh : It’s mathematical formula is $\mathbf{f(x) = 1 - exp(-2x) / 1 + exp(-2x).}$ Now it’s output is zero centered because its range in between -1 to 1 i.e -1 < output < 1 . Hence optimization is *easier* in this method hence in practice it is always preferred over Sigmoid function . *But still it suffers from Vanishing gradient problem.*

Accuracy on Pen digits:

Using sigmoid = 85.52 %
Using tanh = 72.2 %

Accuracy on Dermatology:

Using sigmoid = 91.2 %
Using tanh = 74.40 %

2) Cross Entropy Loss function is used

3) Number of units in Hidden layers

Accuracy on Pen digits:

Number of units in Hidden layers	**Accuracy**

10	60.2 %
20	90.9 %
40	88.5 %

Accuracy on Dermatology:

Number of units in Hidden layers	**Accuracy**
10	65.6 %
20	84.4 %
40	90.1 %

4.Stopping Condition:

Many type of stopping condition can be used.

1) Error falls below some particular threshold.
2) Total change in weights of Neural Network below some particular threshold.
3) Run some particular number iteration.

5. DataSets:

Learn a 3 class classifier for Dermatology dataset for classes psoriasis, seboreic dermatitis, lichen planus. (Divide data in train and test set and report the findings).

Learn 4 class classifier for Pendigit recognition dataset for any four digits. (Use pendigit.tra for training and pendigit.tes for testing)