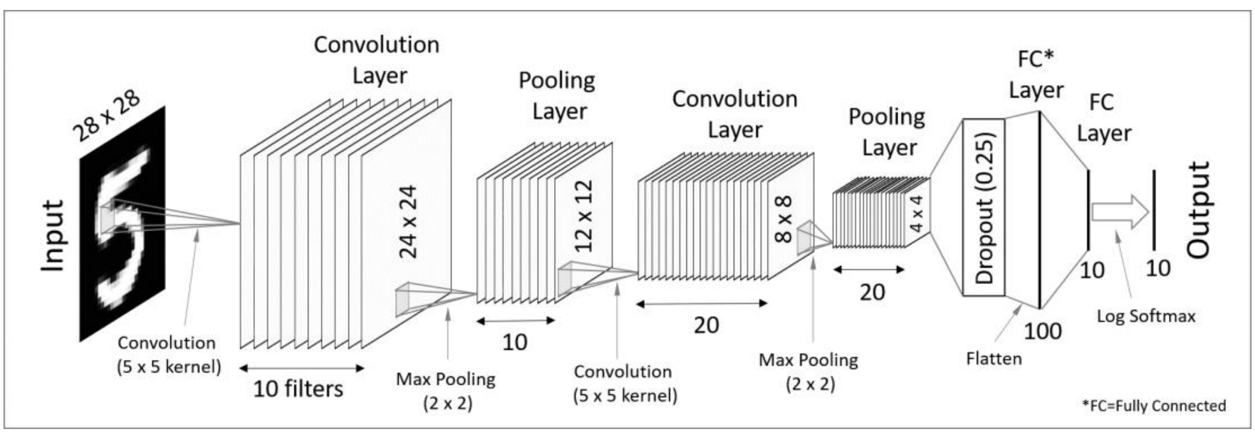
Lecture-5

CNN with an actual MNIST image

Submitted by :- Harish Kumar, Anand Vibhuti and Ashish Kaushal

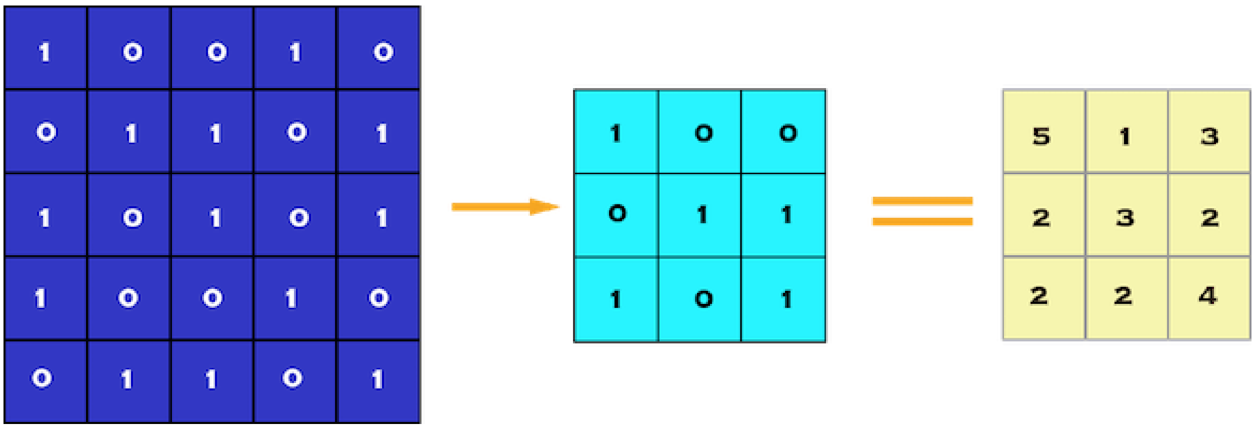
MNIST Image:- The MNIST database (**Modified National Institute of Standards and Technology** database) is a large database of handwritten digits that is commonly used for training various image processing systems. The database is also widely used for training and testing in the field of machine learning. The MNIST database contains 60,000 training images and 10,000 testing images and the dimensionality of each image sample vector is 28 * 28 = 784, where each element is binary.

Architecture of CNN for MNIST image



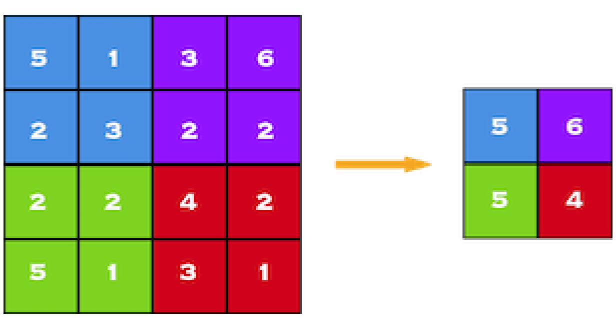
Convolutional Layers

Convolutional layer is the very first layer where we extract features from the images in our datasets. Due to the fact that pixels are only related with the adjacent and close pixels, convolution allows us to preserve the relationship between different parts of an image. Convolution is basically filtering the image with a smaller pixel filter to decrease the size of the image without losing the relationship between pixels. When we apply convolution to 5x5 image by using a 3x3 filter with 1x1 stride (1 pixel shift at each step). We will end up having a 3x3 output.



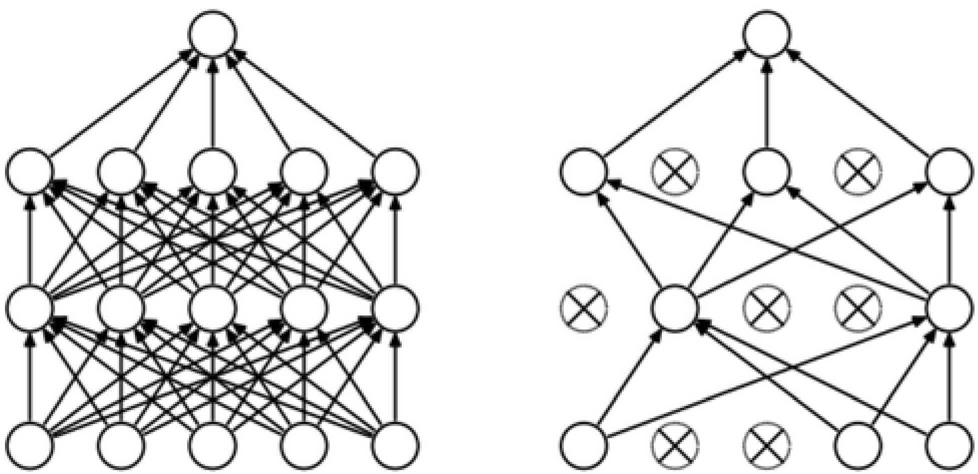
Pooling Layer

When constructing CNNs, it is common to insert pooling layers after each convolution layer to reduce the spatial size of the representation to reduce the parameter counts which reduces the computational complexity. In addition, pooling layers also helps with the overfitting problem. Basically we select a pooling size to reduce the amount of the parameters by selecting the maximum, average, or sum values inside these pixels. Max Pooling, one of the most common pooling techniques, may be demonstrated as follows:



Dropout:

Dropout is a regularization technique used in neural networks to prevent overfitting. For dropout we go through each layer of network and set some probability of eliminating a node in neural network. Eliminating these units at random results in spreading & shrinking of weights.



(a) Standard Neural Net (b) After applying dropout.

Explanation of the working of each layer in CNN model:

Layer 1 is Conv2d layer which convolves the image using 10 filters each of size kernel (5*5).
Layer 2 is MaxPooling2D (2*2) layer which picks the max value out of a matrix of size (5*5).
Layer 3 is again a Conv2D layer which is also used to convolve the image and is using 20 filters each of size (3*3).
Layer 4 is MaxPooling2D (2*2) layer which picks the max value out of a matrix of size (5*5).

Layer 5 is showing Dropout at a rate of 0.25.
Layer 6 is flattening the output obtained from layer 5 and this flatten output is passed to layer7.
Layer 7 is a hidden layer of neural network that is fully connected layer.
Layer 8 is the output layer having 10 neurons for 10 classes of output that is using the softmax function.

So it is a simple feed-forward convolutional neural network (CNN), which takes a 28 x 28 pixel, greyscale, input image, that is then fed through several layers, one after the other, and finally gives an output vector, which contain the log probability (since we will use the Negative Log Likelihood loss function) that the input was one of the digits 0 to 9. Training the network means that you have a dataset of matching input-output pairs. So if you give a hand written digit of a 5 as an input, know what the expected output is, in this case a vector of zeros with a one at index 5 (this is also called one-hot encoding).

Popular CNN Architectures

We achieved accuracy of 99.09% using our simple ConvNet architecture. There are many popular CNN architectures which can be used to achieve better accuracy on MNIST dataset, some of these architectures are:

1. VGG
2. Resnet
3. LeNet-5

Many competitors also use ensemble of these models to get slightly better accuracy. In my next notebook I will discuss an ensemble method with VGG & ResNet to improve our accuracy on MNIST.