**Digit Recognition using CNN Model trained on MNIST Dataset**



The objective of this project is to build a handwritten digit-recognizer using Convolutional Neural Networks to accurately categorize the handwritten digits. For this project, MNIST data is considered for training.

Project is divided into following steps for better understanding :

1. **Downloading the MNIST Data :**

The MNIST dataset is one of the most common datasets used for image classification and accessible from many different sources. In fact, even Tensorflow and Keras allow us to import and download the MNIST dataset directly from their API.

The MNIST database contains 60,000 training images and 10,000 testing images taken from American Census Bureau employees and American high school students.

X\_train and x\_test parts contain greyscale RGB codes (from 0 to 255) while y\_train and y\_test parts contains labels from 0 to 9.

1. **Reshaping and Normalizing the Images data downloaded :**

To check the dataset, for X\_train.Shape , We will get (60000, 28, 28). As you might have guessed 60000 represents the number of images in the train dataset and (28, 28) represents the size of the image.

To be able to use the dataset in Keras API, we need 4-dims numpy arrays. However, as we see above, our array is 3-dims. In addition, we must normalize our data as it is always required in neural network models. We can achieve this by dividing the RGB codes to 255

1. **Building the Convolutional Neural Network**

We will build our model by using high level Keras API which uses either TensorFlow or Theano on the backend. We will import the Sequential Model from Keras and add Conv2D, MaxPooling, Flatten, Dropout, and Dense layers.

1. **Compiling and Fitting the Model**

Now it is time to set an optimizer with a given loss function which uses a metric. Then, we can fit the model by using our train data.

1. **Evaluating the Model**

The results are pretty good for 10 epochs and for such simple model. We achieved 98.5% accuracy with such basic model