**MACHINE LEARNING**

**PROGRAMMING ASSIGNMENT -2**

**NAME: GUNDALA VENKAT ANKIT**

**UTA ID:1002069069**

**MNIST DATASET ANALYSIS:**

The MNIST dataset is a collection of handwritten digits widely used for training and testing image processing and machine learning algorithms.

MNIST stands for Modified National Institute of Standards and Technology database.

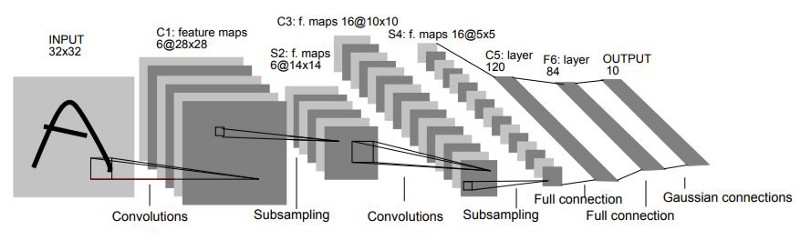
The dataset contains 60,000 training images and 10,000 testing images, each of which is a grayscale 28x28-pixel image of a handwritten digit (0 to 9).

The goal of using the MNIST dataset is to train a model that can accurately classify each image into its corresponding digit label.

The MNIST dataset has been widely used as a benchmark for evaluating machine learning models, and many state-of-the-art techniques have been developed and tested using it.

**ARCHITECTURE OF LE-NET 5:**

LeNet-5 is a convolutional neural network (CNN) architecture developed in 1998. It was one of the first successful neural network models used for handwritten digit recognition.



The LeNet-5 architecture consists of seven layers**:**

* The first layer is the input layer, The input layer takes in the 28 x 28 grayscale image.
* Convolutional layer (C1): The first convolutional layer applies six filters to the input image that are each 5x5, to identify regional features.
* Subsampling layer (S2): The first subsampling layer performs max pooling over 2x2 windows. This reduces the size of the feature maps to 14x14.
* Convolutional layer (C1): The first convolutional layer applies six filters to the input image that are each 5X5.
* Subsampling layer (S4): The second subsampling layer performs max pooling over 2x2 windows in each of the 16 feature maps.
* Fully connected layer (C5): The fully connected layer consists of 120 neurons that are connected to all 5x5 feature.
* Output layer: The output layer consists of 10 neurons, each representing one of the digits from 0 to 9. The SoftMax activation function is applied to the outputs.

**RESULT:**

The model was trained for 10 epochs, and I have achieved up to 98.81% accuracy.

**Table

Description automatically generated**

**ANALYSIS:**

I have trained the model using pytorch. The leNet-5 program is recognizing the input images very quickly. I have trained it for only 10 epochs , which is very low still the neutral network is performing well and accuracy is 98.78%