01

MNIST dataset

Keras API to load dataset

Deeplearning framework – TensorFLow

from tensorflow.keras.datasets import mnist

**Data preparation:**

Flatten the data – 2d array to 1d array

Normalize the data – converting int to floating point btw 0 and 255

Categorize the data - This kind of transformation modifies the data so that each value is a collection of all possible categories, with the actual category that this particular value is set as true.

Keras provides a utility to [categorically encode values](https://www.tensorflow.org/api_docs/python/tf/keras/utils/to_categorical), and here we use it to perform categorical encoding for both the training and validation labels:import tensorflow.keras as keras

num\_categories = 10

y\_train = keras.utils.to\_categorical(y\_train, num\_categories)

y\_valid = keras.utils.to\_categorical(y\_valid, num\_categories)

This is an example of the model learning to categorize the training data, but performing poorly against new data that it has not been trained on. Essentially, it is memorizing the dataset, but not gaining a robust and general understanding of the problem. This is a common issue called overfitting.