Application of NN on MNIST Dataset

Summer Project 2021

Indian Institute of Technology Kanpur

July31,2021

By:

Arkonil Dhar   Saumyadip Bhowmick   Shreya Pramanik

Shubha Sankar Banerjee   Souvik Bhattacharyya

Introduction

Neural Network is a supervised learning model with a huge area of application. It is a non-linear statistical model and can be used for both regression and classification problems. In this project we applied a basic neural network model on the MNIST hand-written image dataset.

Data Description

The dataset consists of 60000 unrotated and 22000 rotated image of hand-written digits.

- The unrotated data is further splitted into 50000 train data and 10000 test data. Each sample is an array of size (784,1). Both the training and the test data is divided into 10 classes for each digits and each class have equal number of samples.
- The rotated data is further splitted into 12000 train data and 10000 test data. Each sample is an array of size (28,28). Both the training and the test data is divided into 10 classes for each digits and each class have equal number of samples.

Model Overview

A neural network is a number of layers stack upon one another where each layer has a number of units. The first layer and the last layers are called the input layer and the output layer respectively. All the other layers are called hidden layers and the units in those layers are called the hidden units. The values of the units are calculated as follows:

If the i^{th} layer of the model have l_i units with values $x_i = (x_1, x_2, \ldots, x_{l_i})'$ then

$$z_{i+1} = W_i x_i + b_i$$

$$x_{i+1} = g(z_i)$$

where W_i is a matrix of order (l_{i+1}, l_i) and b_i is a column vector of order $(l_{i+1}, 1)$ and $g()$ is an activation function. The parameters, W_i and b_i are estimated by the backpropagation algorithm.

In classification problem the number of hidden units in the output layer is taken as the number of classes. As there are 10 digits to classify we took 10 output units and applied the softmax activation function to estimate the probabilities of being classified as the different classes. We used cross entropy loss to calculate the losses in each training epoch and used Adams optimizer to minimize the cost function.

In order to get the best model architechture we make 10 models each with a number of hidden layers ranging from 1 to 10. The number of units in each hidden layer of the models is chosen randomly from the set 16,32,48,…,512. We take 20 such samples for each model resulting with a total of 200 models to train. We trained each model and took the architecture for which the validation accuracy is maximum.

Model Training

To train each model we splitted the train data in 90-10 ratio for training and validation data. We applied early stopping method to avoid overfitting the training data. The best model architecture found in this way had 1 hidden layer with 400 hidden units. Model architechture:

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_11 (Dense)             (None, 400)               314000
_________________________________________________________________
dense_12 (Dense)             (None, 10)                4010
=================================================================
Total params: 318,010
Trainable params: 318,010
Non-trainable params: 0
_________________________________________________________________
```

The model yielded 99.98% accuracy on the train data and 97.91% accuracy on the test data.

Testing on Rotated Data:

The model we built previously was fitted on only unrotated data. Any rotated image of any hand written digit is well outside the distribution of the training data. So accuracy should decrease if we test the model on the rotated data. Unsurprisingly the test accuracy on the rotated data is 34.21%.

Model on Merged Data:

To get a neural network model that will be able to classify both rotated and unrotated image more accurately we combine the two train and test datasets and fit a new model on the combined dataset and test it with the combined test data. We proceed in same way as before except we include a dropout layer with each of the hidden layers with dropout probability selected randomly between 0 and 0.5. The best model for which valdidation accuracy is the maximum is with 5 hidden layers with number of hidden units respectively, 480, 368, 512, 416, 272. Model architechture:

```
Model: "sequential_24"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_144 (Dense)            (None, 480)               376800
_________________________________________________________________
dropout_120 (Dropout)        (None, 480)               0
_________________________________________________________________
dense_145 (Dense)            (None, 368)               177008
_________________________________________________________________
dropout_121 (Dropout)        (None, 368)               0
_________________________________________________________________
dense_146 (Dense)            (None, 512)               188928
_________________________________________________________________
dropout_122 (Dropout)        (None, 512)               0
_________________________________________________________________
dense_147 (Dense)            (None, 416)               213408
_________________________________________________________________
dropout_123 (Dropout)        (None, 416)               0
_________________________________________________________________
dense_148 (Dense)            (None, 272)               113424
_________________________________________________________________
dropout_124 (Dropout)        (None, 272)               0
_________________________________________________________________
dense_149 (Dense)            (None, 10)                2730
=================================================================
Total params: 1,072,298
```

```
Trainable params: 1,072,298
Non-trainable params: 0
_________________________________________________________________
```

The model yielded 99.91% accuracy on the train data and 91.54% accuracy on the test data. The test accuracy is a lot lower compared to the train accuracy indicating a high variance problem. The cause may be the uneven proportion of the rotated and unrotated data in the train and test datasets.