MNIST CNN

Basma Ashour

Abstract
In this paper I'm gonna explain how did I use CNN with Tensorflow/keras for MNIST Dataset and got 99% Accuracy

Keywords
CNN — Tensorflow — Keras

[1] *MIU, Computer Science*

Contents

Introduction

This model is using 7 layers of convolutional neural network and used Adam Optimizer to get a high accuracy, I tried with Delta Optimizer but the accuracy was low so I changed to Adam Optimizer.

1. EDA

First of all I read the MNIST Dataset using keras so I didn't have any local files just used keras Dataset, after that I normalized the data before hands such that large terms of the calculations can be reduced to smaller terms. Like, we can normalize the x.train and x.test data by dividing it with 255. We need 10 classes in output. To make output for 10 classes, use keras utilsTo categorical function, which will provide with the 10 columns.

2. CNN Model

in the CNN model we used 7 layers,
layer1 is Conv2d layer which convolves the image using 32 filters each of size (3*3).
layer2 is again a Conv2D layer which is also used to convolve the image and is using 64 filters each of size (3*3).
layer3 is MaxPooling2D layer which picks the max value out of a matrix of size (3*3).
layer4 is showing Dropout at a rate of 0.5.
layer5 is flattening the output obtained from layer4 and this flatten output is passed to layer6.
layer6 is a hidden layer of neural network containing 250 neurons.
layer7 is the output layer having 10 neurons for 10 classes of output that is using the softmax function.

In the calling and the compile function I used Adam optimizer to increase the accuracy of he model

3. Result

Evaluate function: model.evaluate provides the score for the test data i.e. provided the test data to the model. Now, model will predict class of the data and predicted class will be matched with y.test label to give us the accuracy.

The final Accuracy of the model was 99% which is really good.