
# American International University-Bangladesh

**Submitted by:**

**Name:** Md. Fahad Hossain

**ID:** 18-39211-3

**Section:** B

**Course:** COMPUTER VISION AND PATTERN RECOGNITION

# Implement a CNN architecture to classify the MNIST handwritten dataset


## Abstract:

The purpose of this paper is to construct a convolutional neural network application for the problem of image classification. To assess the performance of the CNN model, I used the MNIST dataset. In the field of computer vision, numerous methods for recognizing images have been created. Our goal is to develop a model that can more accurately determine and identify handwritten digits from the MNIST dataset. I used an activation function, an optimizer, and a learning rate to do this experiment. Adam, SGD, and RSMProp are the three types of optimizers I utilize. The precision of the optimizer rate or epochs is affected. Results are displayed in the activation function. Using these three optimizers, we get a 98 % accuracy rate.


## Introduction:

A convolutional neural network is a class of artificial neural network, most commonly applied to analyze visual imagery. CNNs are powerful image processing, artificial intelligence that use deep learning to perform both generative and descriptive tasks, often using machine vison that includes image and video recognition, along with recommender systems and natural language processing

CNN consists of many layers like-the convolutional layer, the pooling layer, the ReLU correction layer and the fully-connected layer. The main aim of convolutional layer is to detect the presence of a set of features in the images received as input. The principle is to “drag” a

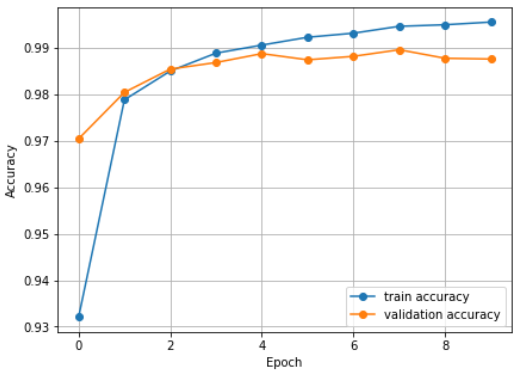
window representing the feature on the image, and to calculate the convolution product between the feature and each portion of the scanned image. A convolutional layer can detect more features depending on the features that we provide to it. Here I have used three types of optimizer Adam, SGD, RSMProp.

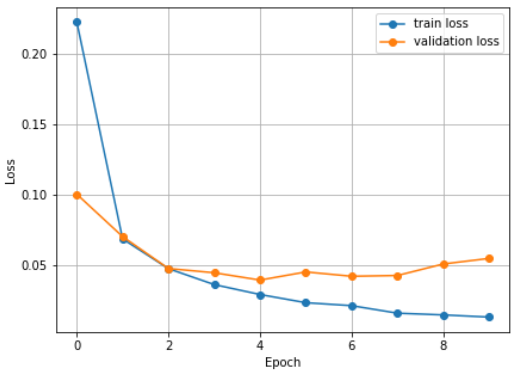
The Adam optimization algorithm is an extension to stochastic gradient descent that has recently seen broader adoption for deep learning applications. This algorithm can be used instead of the classical stochastic gradient descent procedure to update network weights iterative based in training data. ADM optimizer can handle sparse gradients and also, we can obtain the best accuracy.

Stochastic gradient descent (SGD) is an iterative method for optimizing an objective function with suitable smoothness properties. It can be regarded as a stochastic approximation of gradient descent optimization, since it replaces the actual gradient by an estimate thereof.

RMSprop (Root Mean Squared Propagation) is a gradient-based optimization technique. This normalization balances the step size (momentum), decreasing the step for large gradients to avoid exploding and increasing the step for small gradients to avoid vanishing and uses decaying average of partial gradients.

## Results:



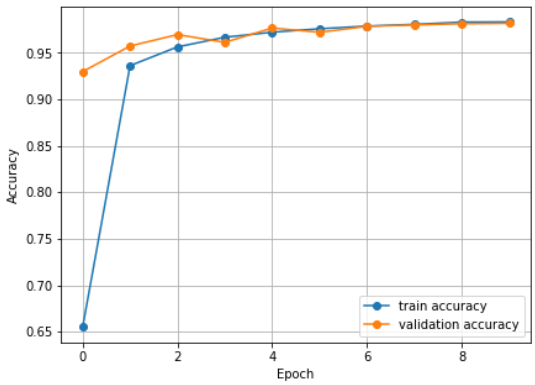



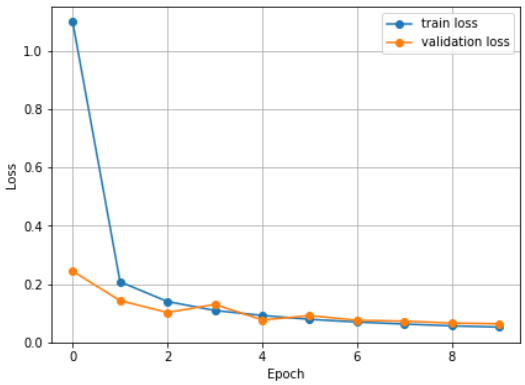

```
[9]  test_loss, test_acc = model.evaluate(x_test, y_test)
     print('\nTest Accuracy:', test_acc)
     print('\nTest Loss:', test_loss)
```

```
313/313 [==============================] - 3s 11ms/step - loss: 0.0469 - accuracy: 0.9883

Test Accuracy: 0.9883000254631042

Test Loss: 0.04694044217467308
```



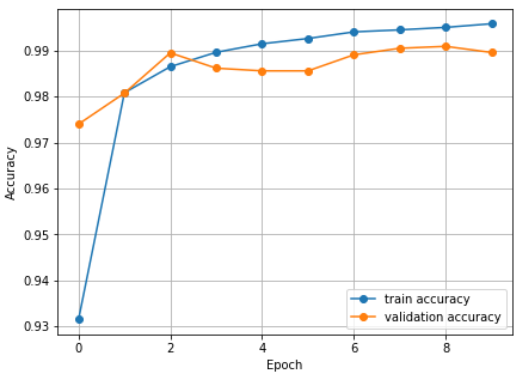



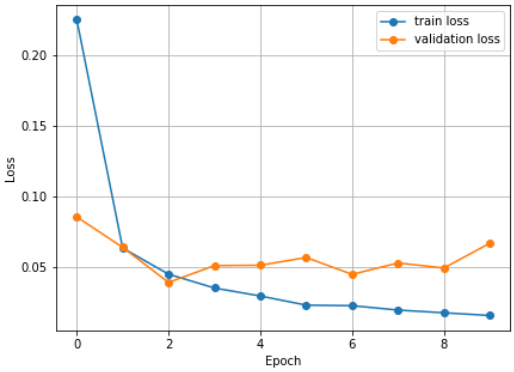

```
[11] test_loss, test_acc = model.evaluate(x_test, y_test)
     print('\nTest Accuracy:', test_acc)
     print('\nTest Loss:', test_loss)
```

```
313/313 [==============================] - 1s 4ms/step - loss: 0.0527 - accuracy: 0.9829

Test Accuracy: 0.9829000234603882

Test Loss: 0.05268304422497749
```




```
[27] test_loss, test_acc = model.evaluate(x_test, y_test)
     print('\nTest Accuracy:', test_acc)
     print('\nTest Loss:', test_loss)

     313/313 [==============================] - 1s 4ms/step - loss: 0.0552 - accuracy: 0.9899

     Test Accuracy: 0.9898999929428101

     Test Loss: 0.05523671954870224
```

## Discussion:

Here we have obtained the results using the three types of optimizers in order to solve image classification problem on MNISY hand written dataset. ADAM optimizer is used at first which gave 98.83% accuracy for MNIST dataset. Using SGD, the accuracy obtained is 98.29% and test loss is 5.27%. Lastly from RSMProp optimizer test accuracy obtained is 98.99% and test loss is 5.52%. From here it is visible that maximum accuracy is obtained from RMSProp optimizer. On the other hand, using SGD minimum accuracy is obtained. Therefore, we can easily get faster and high accuracy from RMSProp optimizer compared to ADAM and SGD optimizer.