

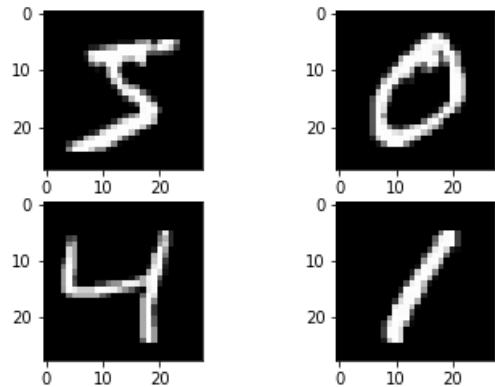
Assignment 2.1 CNN and Pre-Trained Models

The goal of this assignment is to work on Convolutional Neural Networks. Please implement the coding parts and clearly answer the questions.

Part I CNN

1- For the first part of the assignment, we want to work on a black and white image which is an OCR (Optical Character Recognition). Optical character recognition or optical character reader is the electronic or mechanical conversion of images of typed, handwritten, or printed text into machine-encoded text, whether from a scanned document, a photo of a document, a scene-photo or from subtitle text superimposed on an image <Wikipedia>.

1.a) Start with creating a visualization of your input data.

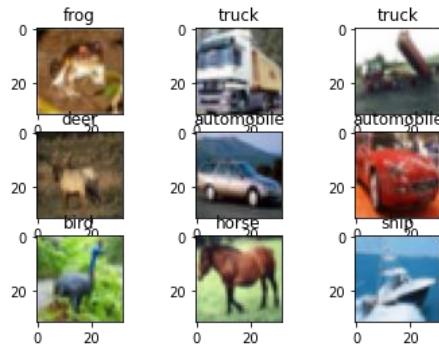


1.b) Create a CNN model with 4 convolution layers in which two of them have 32 and two of them have 64 filters. The fully connected layer has one hidden layer (512 nodes). Draw the Learning curve. What is your understanding of the learning curve? Batch size=128 and epochs=20.

Part 2 CIFAR-10

2- For the next part we work on a CIFAR-10 Dataset. This is a dataset of 50,000 32x32 color training images and 10,000 test images, labeled over 10 categories. You can get more information about the dataset here: [CIFAR-10 and CIFAR-100 datasets \(toronto.edu\)](http://www.cs.toronto.edu/~kriz/cifar.html). The difference this time is that this is a colorful dataset.

2.a) As before, start with visualizing your data to get some insights. Plot the first 9 images from the dataset.



2.b) Apply the pre-processing algorithms that we discussed last week. The augmented images are supposed to be scaled by 20%, zoomed by 20%, and horizontally flipped. Now, design a CNN model with 4 convolution layers in which two of them have 32 and two of them have 64 filters. The fully connected layer has two hidden layers (512 and 256 nodes respectively). Draw the Learning curve. What is your understanding of the learning curve? Batch size=128 and epochs=20. Apply the model on the processed data.

For the optimizer, use Stochastic gradient descent with a learning rate of 0.005, decay of 1×10^{-6} , momentum of 0.9.

2.c) Recommend a technique to resolve the issue that you saw on 2.b and implement your recommendation.

2.d) Apply at least two pre-trained models on your data and draw the learning curve. What is the test accuracy? How do you compare the models?