Homework 5: Deep Learning

Out May 30; Due June 5, 12 a.m.*

Kristian Kersting, Dominik Hintersdorf, Quentin Delfosse
{kersting, dominik.hintersdorf, quentin.delfosse}@cs.tu-darmstadt.de

In this homework, you will again use a neural network to classify MNIST.

1 Convolutional NN in Pytorch

We have discussed during the last lectures that convolutional neural networks (CNNs) are inspired by the cat's visual cortex. CNNs have been very successful on a variety of learning problems, including, but not limited to, image classification.

The goal of this task is to implement a CNN using deep learning frameworks such as TensorFlow, PyTorch (preferable) and MXNet. For this task, we do not provide a structure for the solution. Please fill out the provided, empty notebook file with your own code.

By now, a lot of CNNs have been proposed in the literature, but we will focus on the most simple CNN architecture. LeCun et al. [1998] have proposed LeNet that is composed of pairs of a convolution layer and subsampling layer, followed by fully connected layers, illustrated in fig. 1.

As we use MNIST also for this task, that is, inputs are gray-scale (single channel) images, the actual input size of your CNNs should be 32×32 if the kernel size is 5×5 and stride is 1. In other words, MNIST digit images (28×28) are padded with zeros around the border, so that inputs are of 32×32 matrix. If you want more examples about convolution operations and zero padding, please go check the following web page: `https://github.com/vdumoulin/conv_arithmetic`

In the subsampling layers, the kernel size is 2×2 and stride is 2. Please note that subsampling in LeNet is mean (average) pooling. Your first CNN architecture should have the same number of parameters as LeNet-5.

*We will discuss the solutions in the exercise session. It is my suggestion that you try to address at least 50% of the exercise questions. Simply try hard to solve them. This way, you will get familiar with the technical terms and with the underlying ideas of the lecture.

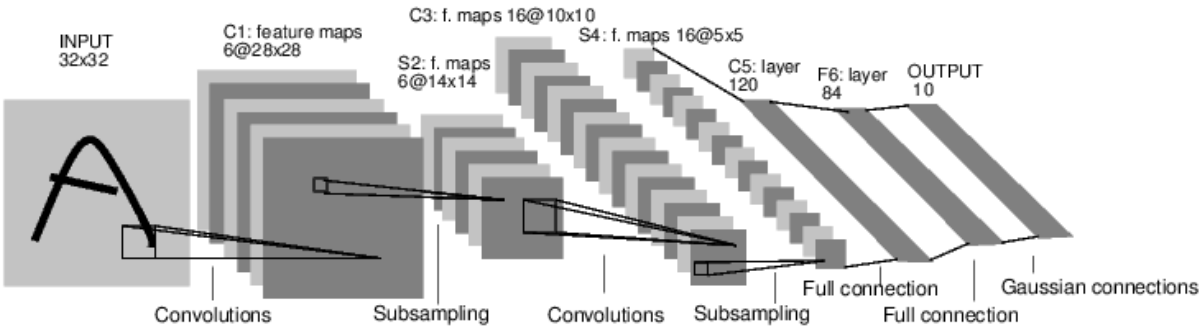

Figure 1: Architecture of LeNet-5, taken from LeCun et al. [1998]

Although you are supposed to use deep learning frameworks, please note that your task is to implement ***your own convolution layer and pooling layer***. **DO NOT** use the convolution layer class and pooling class provided in such a deep learning framework.

1. Thus, your task is to provide an implementation of LeNet-5, by implementing your own convolution and pooling layer.

2. Write a training procedure for your LeNet-5 implementation for MNIST digit classification.

References

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.