NNDL: Homework 1
Supevised Deep Learning

Alessandro Lovo

January 5, 2021

1 Introduction

This homework consists in applying supervised deep learning to two tasks: a regression task consisting in approximating a scalar function of a scalar variable and a classification task that is recognizing the handwritten digits of the MNIST dataset. For the regression task a fully connected network (FCN) will be used, while for the classification task both a fully connected but most importantly a convolutional network (CNN) will be tested. In both cases different architectures, optimization and visualization techniques and hyperparameter search will be tried.

1.1 General framework

Both tasks rely on a framework of python classes that unfolds as follows:

- **Net** class inheriting from *torch.nn.Module* that contains the actual neural network with a specific architecture.
- **Evolver** class for handling the training and validation of a *Net*. In this class there is a check at the end of every training epoch to interrupt the learning process. To implement early stopping one just needs to inherit from the *Evolver* class and specify that check condition. In particular the learning process stops if the validation loss isn't decreasing after *patience* number of epochs.
- **KFoldCrossValidator**: class for performing k fold cross validation on a particular set of hyperparameters.

2 Method

3 Results