097200- Deep learning: Warm-up exercise

Submission date: 03/12/2019

1. Exercise

In the following exercise, you will create a classifier for the **Fashion MNIST** dataset. You should write your own training code and meet the following constraints. You can submit your work only after achieving the performance levels listed below.

The goal of this exercise is to get familiar with training a simple neural network. You should try different hyper parameters (model architectures, learning rate, batch size, activation functions etc.) to see how it affects the performance.

2. Submission instructions

Submission will be in pairs (course partners) and will contain a short (two pages) pdf report containing:

- Model architecture description and illustration, training procedure (hyper parameters, optimization details etc.).
- Two convergence plots - (1) for error (2) for loss, both as a function of time (epochs). Each plot should depict both training and test performance (i.e. two curves per plot, one for train and one for test).
- A short summary of your attempts and conclusions.

 In addition, you should also supply:

- Code (python file) able to reproduce your results - we might test it on different variants on these datasets.
- The trained network with trained weights (.pkl file).
- A function called *"evaluate_hw1()"*. The function should load the FashionMNIST test-set, load your trained network (you can assume that the data and model files are located in the script folder) and return the average accuracy over the test-set. (accuracy = 1 - error)
 This function should be written in a separate script.

   Moodle submission:

   you should submit a Zip file containing:

- **2 python files (training procedure, evaluation)**
- **1 pdf file (containing your names and ID's)**
- **.pkl file (If the file is too big for the Moodle, upload it to your Google-Drive and copy the link to your pdf report)**

3. Classification Network

Write a complete training procedure for a classification network on FashionMNIST dataset. Design and train your network so that it will satisfy the 2 following goals:

- The network should consist of linear layers and activations (without convolutions).
- Number of trainable parameters (weights) within the network should be **< 65,000**.
- Final accuracy on the test-set should be **> 80%** (not too hard… try to get as close as possible to 100%)

4. Data

You will train the network ONLY on torchvision.datasets.FashionMNIST with train=true, and test the network with train=false.

5. Grades policy

- Successful submission - 50 points.
- Report - 15 points.
- Competition - 35 points: the teams will be sorted according to their test error results, then will be split into groups of 5. The group with the best performance will receive $35 - 0*(35/\#groups)$, the next best group will receive $35 - 1*(35/\#groups)$ and so on.

Good luck!