Laboratory exercise: convolutional neural networks

In this lab exercise you are going to study the performance of different CNN configurations when used to classify images. The data set proposed is the *CalTech 101 Silhouettes Data Set*, which is described and can be downloaded in the following link: https://people.cs.umass.edu/~marlin/data.shtml. The data set is named "caltech101_silhouettes_28.mat".

The set of images available represents 101 different silhouettes (e.g. bonsai, chair, elephant, etc.). Therefore, you have 101 possible output classes. On the other hand, each image is represented by 28x28 pixels, which corresponds to 784 classification features (input variables). The number of instances, i.e. number of images, available for this problem is 8671.

The idea is to study the performance of different parameter configurations for a CNN neural network. In order to compute and report the performance of each configuration it is required to use the mean accuracy measure. Each configuration should be executed at least 3 times in order to get the mean accuracy value.

The base architecture is a CNN with the following (and sequentially connected) structure:

1) An input layer.
2) A number **NB** of convolutional blocks composed of
   2.1) A convolutional layer (kernel size=3, filter size=**FS**).
   2.2) A non-linear hidden layer (**NHL**).
   2.3) A max-pooling layer (size=2, stride=2).
3) A fully connected layer (**FC**).
4) A non-linear output layer (**OL**).
5) A cost function layer (**CFL**).

You should, at least, study the following configurations:

1) Compare the performance of a CNN composed of one convolutional block **NB**=1 (**FS** = 128) wrt a CNN composed of three convolutional blocks **NB**=3 (**FS** = 32, 64 and 128, respectively).
2) Compare the performances of the Sigmoid and Rectified Linear activation functions in **NHL.**
3) Different percentage of training, validation and test data sets: 80/10/10, 40/20/40 and 10/10/80.

Remember to encode the labels accordingly for every configuration. It is recommended to use the Adam training algorithm in order to save computational time. The rest of general parameters (activation function in the non-linear output layer **OL**, cost function **CFL**, maximum number of epochs, L2 regularization parameter - if used, etc) and specific parameters (learning rate and momentum for SGD and decay factors for Adam, etc) can be set a priori, but you have justify your choice or perform a previous search to find reasonable values. Additionally (and optionally), you can test other architectures, perform any modification to the base architecture described or use another training algorithm.

Write a brief document (four sheets maximum) that includes:

1) Description of the runs with the different configurations that you have performed.
2) Explain how you have selected the rest of parameters.
3) Those tables that you consider necessary to describe the results obtained for the different network configurations. Explain and reason the results presented in the tables.
4) Your own conclusions with respect the results obtained.
5) Include the *.m* files with your code.
6) **If you use ChatGPT (or another similar tool) in the document, indicate it every time it is used. We want to evaluate your work, not someone else's.**

IMPORTANT: It is always a good idea to test the hardware+software environment with enough advance to make a good estimation of the execution times. In particular, for this exercise you will see a big difference when running the experiments on a computer equipped with a GPU.

NOTE: The exercise is designed to be done in Matlab, but you can implement it in another environment.

REMEMBER that you can find the basic instructions for the construction and training of a CNN with Matlab in the Lab Class Guide