SGN-13006 Introduction to Pattern Recognition and Machine Learning
TAU Computing Sciences
Exercise 4 *Visual classification with NNets (CIFAR-10 dataset)*

Be prepared for the exercise sessions. You may ask TA questions regarding your solutions, but don't expect them to show you how to start from the scratch. Before the end of the session, demonstrate your solution to TA to receive exercise points.

1. **CIFAR-10 – Neural Networks** (60 points)

   Data: We will use the CIFAR-10 dataset.

   Neural network training works the best if you use a single binary output for each class and the classification is the maximum of them. You need to change your label data as following:

   $$0 \rightarrow (1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0)$$
   $$1 \rightarrow (0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0)$$
   $$2 \rightarrow (0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0)$$
   $$\ldots$$

   MLP training: There exists a Graphical User Interface of Matlabs Neural Networks Toolbox (type `''>> nnstart'')` which you can use to find good network parameters. Note, however, that under the GUI there are Matlab functions that you can directly use - see the Matlab documentation by typing: `''>> doc patternnet''`.

   By changing the "net" Matlab structure you can change the number of neurons, training functions etc.

   Note: Matlabs Neural Networks Toolbox expects both input and output data to be column vectors.

   Write a function *net=cifar_10_MLP_train(trdata,trlabels)* that trains an MLP classifier for the CIFAR data. Write also *estlabel=cifar_10_MLP_test(x,net)* that estimates the label for the given input x (can be multiple rows).

   Train a classifier and evaluate its performance for the CIFAR-10 test data. Compare the classification accuracy to the previous methods.

   Note that as the input features you can use any of those used in the previous experiments (mean value of each channel or sub-window based mean values) or you may use the whole image as the input.

   Python: You may use the code showed during the lectures as the starting point (you need to install Python to your computer)