1. Introduction

Imagine that you are a member of a research group in a company. Your group leader has asked you to **understand a training algorithm fully by implementing the method, testing it out, explore its behaviour and explaining it, and perhaps even suggesting ways it might be improved.**

To do this assignment you will need a copy of the paper by **Mosca**, entitled "Training Convolutional Networks with Weight–wise Adaptive Learning Rates", published in the ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 26-28 April 2017. **You will implement the training algorithm described in the paper, called WAME, and you will test it on a real-world data set. You have to describe how the method works, adjust its free parameters, discuss your experimental set up and explain your experiments. Lastly, you have to evaluate how the algorithm solves the problem, and discuss your results and their significance.**

You can use any programming language or software library for this assignment (**Jupyter - Python**). In the labs, we use MATLAB which provides well tested functions for neural networks design.

2. Implementation and experimentation

You can do your WAME implementation in MATLAB, write your own code or build on a package/library from the internet. You are not tested on programming so the coding style does not have to be perfect and your code does not have to be optimal but it should obviously work correctly. Since the focus is on implementing WAME, I wouldn't recommend implementing all the methods required for training neural networks, e.g. backpropagation, derivatives calculation etc., from scratch unless you are very experienced with Java, C++, Python or some other programming language or platform. No matter what you do/use, make sure that all sources and code taken by others or the internet are cited properly in your Report; otherwise, you may be accused of plagiarism.

Some packages provide techniques for determining the optimal structures of machine learning models (e.g. the model architecture or the model hyperparameters) automatically as part of the training. In that case, instead of performing experimental tests varying the number of free parameters of the model, these techniques can be used to find the appropriate structure for your model. Still some of these methods may have their own parameters, which require fine tuning.

Note that the performance results of your approach would be more meaningful, if a validation technique is used, such as k-fold cross validation (k=7 or k=10 is typically used), or leave- one-out cross validation, or some form of Monte Carlo simulation. Lastly, the use of regularisation, provided in some software packages and in Matlab, normally helps to get better results.

You are expected to test your implementation of WAME using relevant data and store results in ASCII format for each experiment that you conduct. Results are typically in the form of: number of successfully recognised patterns per class for each experiment; number of unsuccessfully recognised patterns per class for each experiment; the overall average classification success in training and in testing, and average error in training and in testing.

The results of your experiments should be stored in ASCII format and IPYNB format, in a Jupyter notebook.

3. Assignment outline and marking scheme

Your work will be presented in a Report (notebook documents are not accepted as a Report). It is important that your Report is properly structured. Sections like the ones shown below could be included in your report to ensure good coverage of the topic. A number of 2000-3000 words are expected to cover in sufficient depth all aspects of the assignment, but our marking is not based on the number of words used in the Report. Also, you are not just being marked on how good the results of your neural model are. **What primarily matters are that you describe your design of the training algorithm, justify any choices you made, explain how things work, and make the model work with data. Also, you will need to provide insight on how to (pre)process the data before feeding into the model and do the training (if necessary), how to debug the learned model (not only the training algorithm), how to measure model performance and demonstrate its significance.**