ELV832: Assignment, Part 1

Weightage: 15% of course grade
Deadline: 28th March 2025

The basic goal is to replicate the left subfigure of Figure 2 in Barbier et al. 2019, initially only for the logistic regression model (the blue squares in that figure). All your code for the below steps should be clearly organised and commented. Please use Python and in particular the **scikit-learn** library for the logistic regression.

1. Take D=10,000, and generate a random vector of binary weights, i.e., +1/-1. This constitutes your *teacher* model.
2. We need to generate a training set of size 2D (to allow for sample complexity to go up to 2), plus some test data which we can keep of size 0.2D. Hence, generate an overall data set of size 22,000. All feature values should be sampled IID from a standard normal distribution (zero mean, unit variance), and then each feature vector should be passed through your teacher model (with sign() activation for the perceptron) to obtain its corresponding class label.
3. Now, you have to train different logistic regression models using scikit-learn, for different proportions of the training data set corresponding to different sample complexity values. Use values at intervals of 0.2, ranging from 0.2 to 2. Of course, the test set should never be touched during training. For each logistic regression model trained, the regularisation hyperparameter should be finetuned via cross-validation: use scikit-learn to automate this. Finally, each finetuned model is to be tested on the held-out test set, and the error plotted against the sample complexity, just as in the original plot.

These steps complete one basic replication. Now, try at least the following variations.

A) Repeat the above for one more teacher model, i.e., a different set of random binary weights. Compare the results to the initial ones.
B) Repeat the above for a larger value of D=100,000. See how feasible this is in terms of computational time/memory. If possible, make D even bigger, and report the results of the above for the largest value of D you can manage. You can also use the IITD HPC [https://supercomputing.iitd.ac.in/]; you may apply for access via this webpage and mention the instructor's name as the approving faculty member.

Plot all your results clearly and comment on them and what they might be showing you. Are you seeing any signs of the phase transition to perfect generalisation that should theoretically happen? If so, at what value of sample complexity does it occur? If not, why might it not be happening? What might be a way to change the settings of the experiment so as to be able to observe the sharp transition? If you can propose some concrete ideas, you will have a chance to try those out in the second part of this assignment.

You should prepare a brief report with all your results and discussion along the above lines, and that along with all your code will need to be submitted via Moodle, by the deadline indicated on top.