COMS 3007: Machine Learning Assignment 2024
Part 2: Solving ML Problems

You have now studied many different machine learning algorithms, with an emphasis on classification. In this assignment, you will be testing out your ability to tackle classification problems.

You have been given a dataset spread over two files. *traindata.txt* contains a set of input data (one data point per row), and *trainlabels.txt* are the corresponding labels (from 0 to 20).

Your task is simple: you are required to build a classifier to classify unseen data drawn from the same distribution as the training data. You can use any method you want, but your goal is to perform as well as possible on the unseen data. This will be auto-marked, with marks assigned based on your accuracy on this data. Specifically, the marking will be competitive with a live leaderboard, so use everything you have learned to do as well as possible.

You must submit a python file that loads a file of input data points, loads a pre-trained model you have trained and uses this model to predict the labels for each data point which is then saved to an output file. The python file must be called '*classifyall.py*' which reads from a file called '*testdata.txt*' and writes to a file called '*predlabels.txt*'. '*testdata.txt*' will have the same format as '*traindata.txt*' and '*predlabels.txt*' should have the same format as '*trainlabels.txt*'. If it does not, then the marker will give you zero.

In addition to the Python standard libraries *classifyall.py* can only use the following libraries and versions:

(1) numpy==1.26.4
(2) scikit-learn==1.4.2
(3) pandas==2.2.2
(4) torch==2.3.0+cpu *
(5) tensorflow-cpu==2.16.1 *

* Note: GPU is NOT supported for PyTorch or TensorFlow.

You will be provided with the singularity environment that the marker will use so that you can ensure your code will run correctly on the marker. Thus, if your code does not run correctly on the marker you will automatically be given zero for this assignment. Instructions on how to download and use this singularity environment will be provided separately on Moodle.

You are allowed (encouraged) to pre-process your data if it improves accuracy, but you must ensure that '*classifyall.py*' performs the same pre-processing.

You must make two submissions (each just by one person in the group): your code to be auto-marked, and a short report.

Your report should be *at most TWO pages* submitted as a **PDF document** to Moodle. This should include the following points, but should be kept very brief:

(1) All your names and student numbers.
(2) The algorithm you used, with any design decisions, and hyperparameters. Briefly justify what you did, and reference any external papers/websites that you relied on heavily. This document needs to convince me that you understand and can explain exactly how your algorithm works.
(3) Any clever tricks that you found to be helpful, including data pre-processing.
(4) A measure of your expected accuracy (based on any test data you separated out).

Important:

- The due date for submissions is the end of **24th May.**
- **You must submit and work in groups of between three and four people (having more or fewer people will automatically halve your marks). Make sure all your names AND student numbers are on the submission, otherwise you will receive 0.**