**COMP 5212 Machine Learning (2024 Spring)**
**Project**
**Hand out: Mar 30, 2024**
**Due: May 3, 2024, 11:59 PM**
**Total Points: 100**

Your report should contain below information at the top of its first page.

1. Your name
2. Your student id number

Some Notes:

- **Late Policy:** 3 free late days in total across the semester, for additional late days, 20% penalization applied for each day late. **No assignment will be accepted more than 3 days late**.
- Please refer to the Course Logistics page for more details on the honor code and logisitcs. We have zero tolerance — in the case of honor code violation for a single time, you will fail this course directly.

# 1 Classification Task (70 pts implementation + 30 pts report)

In this project, you are given a classification dataset, which contains 7 features and 1 binary label as the last column. The data can be downloaded from canvas, and it has been splitted into train set and validation set, but you are free to choose how to use them (i.e., use them all to train your model). Based on this dataset, you are required to build two models, one traditional machine learning model and one neural network model to do this classification task. Below are some requirements and suggestions for each model.
In addition, you are required to summarize your results, findings, any explorations into a simple report. A simple report should not only discuss the changes made in the model hyper-parameters and the corresponding result, but also analyze the results and why you do such changes.

## 1.1 Machine Learning Model (35 + 15 pts)

Requirements and Suggestions:

1. You can only use traditional machine learning model is this part, which including but not limited to logistic regression, clustering algorithm, support-vector classification, decision tree, random forest...

2. You are not required to build the model from scratch, which means packages like *scikit-learn* is allowed in this project.

3. You are encouraged to think which machine learning best fit this task and apply model selection knowledge you learned from class to see which model and corresponding hyper-parameters are good.

4. You are also encouraged to try regularization, kernel... or more advanced technique from online to see whether it can improve the results.

## 1.2 Neural Network Model (35 + 15 pts)

Requirements and Suggestions:

1. You can only use deep learning model in this part, which including but not limited to FNN, CNN...

2. For fair comparison, the total number of parameters cannot exceed 100,000, otherwise it would be zero point.

3. For training deep learning models, you can use Google Colaboratory (recommended) if you do not have computing resources.

4. You are encouraged to try different techniques in deep learning such as data augmentation, regularization, dropout...

5. Except for the .ipynb file for entire train process and valid process, you must submit the model.py exactly the same with our sample code, to ensure that you could run the grading.py file without any problems since we would use this script to grade your result.

## 2 Submission

1. For machine learning model, you are required to encapsulate your final model into a class called *MachineLearningModel* and submit a py file that contains your model class.

2. For deep learning models, you are required to submit your notebook and python file together with your trained pytorch model checkpoint. Similar to machine learning model, you are requried to encapsulate the model into a class. And if you want to do any data-preprocessing, you need call your data preprocessing function before return the prediction.

3. For both models, we have provided our evaluation script on canvas, the test data has the exact same format as the training/validation data. You are required to make sure this script can run successfully with environment *python* $\geq$ *3.9 , pytorch* $\geq$ *2.0 and scikit-learn = 1.4.1*. For more details about above three points, you can directly refer to the evaluation scripts.

4. In addition to implementation files, you are required to summarize your findings, explorations as a pdf report.

## 3 Grading Policy

The grading will based on two parts: implementation (70 pts) and report (30 pts).

1. For implementation: we have preserve a hidden test set to measure your model's performance and generalization ability. As a reference, the baseline accuracy of machine learning model and neural network model are both 90%. Baseline accuracy is considered easy and if you reach this line, you can get 60% score (42 points). you can earn 10% (7 points) more by achieving 1% higher accuracy. The corresponding validation accuracy on our provided validation set is 88% (validation accuracy is only for your reference, not grading). But do remember, **a higher validation accuracy may not result in a higher test accuracy**.
   If you achieve an accuracy below baseline, we will mark it case by case by evaluate how much effort you put.

2. For report: In case you have done much exploration but the final performance is not very good, we will also evaluate your report. A simple report should not only discuss the changes made in the model hyper-parameters and the corresponding result, but also analyze the results and why you do such changes. The report grading will be based on the amount of work as reflected in your report, coherence of the results, insightfulness of discussions.