THE HONG KONG UNIVERSITY OF SCIENCE & TECHNOLOGY
Department of Computer Science and Engineering
COMP4211: Introduction to Machine Learning
Spring 2022: Assignment 1
Due time and date: 11:59pm, Mar 14 (Mon), 2022.

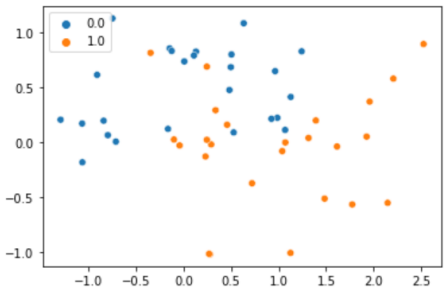
Q1. Consider the boolean function $f(a, b, c, d, e) = (a \vee b) \wedge (\neg b \vee c \vee \neg d) \wedge (d \vee \neg e)$, where a, b, c, d, e are boolean variables (i.e., with values in $\{0, 1\}$).

(a) Draw the truth table of $f(a, b, c, d, e)$, and print it out in assignment1.ipynb.

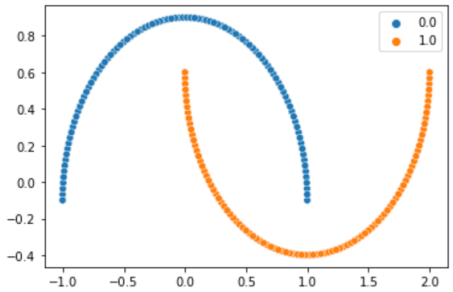
(b) While a single perceptron cannot represent this function, it can be represented by using *a set of* perceptrons. Implement your solution in PyTorch. You can find the weights in this set of perceptrons either manually or by some learning procedure. The activation function should be the step function, as in the lecture slides.

(c) Run your model on all the 32 rows of the truth table in part (a), and show the results in assignment1.ipynb.

Q2. In this question, we will use the so-called "2-moon" dataset. The training dataset (in variable train_loader) is noisy (figure 1a), while the testing dataset (in variable test_loader) is clean (figure 1b). You are also provided with part of the code, in assignment1.ipynb.



(a)



(b)

- Fill in the provided build_mlp() function. The function should take in a parameter nbr_hidden_nodes, and should use PyTorch to build a Multi-layer perceptron (MLP) that has a single hidden layer with nbr_hidden_nodes hidden units. Use ReLu as the activation function of MLP. Lastly, return the built MLP.
- Fill in the provided train_mlp() function. It should train the MLP on the training dataset.
- Use the code to build and train 12 MLPs. These 12 MLPs should use each of $[3, 4, 5, 6, 7, 15, 20, 50, 100, 200, 500, 1000]$ as the number of hidden units. Plot both the training and testing accuracies of MLPs over different numbers of hidden units. For the plot, use the number of hidden units as x-axis, and accuracy as y-axis.

Submission Guidelines

Please submit a completed Python notebook file (based on the assignment1.ipynb file) to show your work. Name the .ipynb file in the format **YourStudentID_assignment1.ipynb** (e.g., 12345678_assignment1.ipynb) and upload it to Canvas. Required results should be shown clearly. **Plagiarism will lead to zero point on this assignment**.