Homework Assignment 5 (Programming Only, 100 points)

You choose **only one** of the following tasks.

There will be no step-by-step instructions, and you need to write the code from scratch using Jupyter notebook (on your computer or google colab).

Write comments in your file, so that I can understand what you have done.

Programming bugs, which cause your program to crash, are not acceptable. If there is a programming bug in your file, you will lose a random number of points. Note: "warning" is not an error.

You may use multiple Jupyter notebook files: one file for one sub-task

Task-1: Transfer learning for COVID-19 diagnosis using CT images (100 points)
You will develop a 2D CNN for COVID-19 diagnosis using CT images.
I will provide you the dataset and the dataloader in Pytorch
(1) 10 points: construct a CNN by modifying Resnet-18 or Resnet-50 for binary classification
(2) 25 points: train the CNN from scratch
(3) 25 points: train the CNN using transfer learning
Compare the two models in (2)&(3) to see if transfer learning brings extra benefits or not.
Note: there are three datasets, training set, validation set, and test set.
test accuracy of (2) should be > 90%, otherwise you will lose 10 points
test accuracy of (3) should be > 90%, otherwise you will lose 10 points
(4) 40 points: visualize the two models (2)&(3) using **two CAM** methods (e.g., GradCAM and EigenCAM) in
https://github.com/jacobgil/pytorch-grad-cam
https://github.com/jacobgil/keras-grad-cam
https://www.kaggle.com/code/nguyenhoa/dog-cat-classifier-gradcam-with-tensorflow-2-0/notebook#GradCAM-&-GuidedGradCAM-class-defnition
or other github repo (cite the link in your code file)

Task-2: DNN for ECG signal Classification on the CPSC dataset (100 points)
read https://github.com/SarielMa/Robust_DNN_for_ECG

You can use the code in this github repo for data processing and data loading.

(1) 60 points: Design a **new DNN** for ECG signal classification on the CPSC2018 dataset, and it must be different from those in the github. The DNN could be CNN, RNN, Transformer, etc, and it cannot be a simple MLP. This is more challenging than the task in homework 4.
Note: test accuracy should be > 75%, otherwise you will lose 20 points
(2) 40 points: **visualize** the DNN model using **two CAM** methods in
https://github.com/jacobgil/pytorch-grad-cam
https://github.com/jacobgil/keras-grad-cam
or other github repo (cite the link in your code file)
Note: you may need to modify the code of CAM because the input is not 2D image.

Task-3: Apply what you have learned to a new dataset in your field of study

You need to send your proposal to me by email and get written approval from me.
It should be not too easy and not too hard, compared to Task-1 and Task-2. If you really want to choose an easy project, then we will have to reduce the total number of points (e.g., 100 points to 60 points)
Note: You may use Kaggle projects only if your model/method will be different from those on Kaggle
https://www.kaggle.com