# CS7304H: Statistical Learning
# 2022 Fall Project

**December 4, 2022**

## 1 Introduction

In this project, you are required to complete a classification task with noisy data. The data come from an anonymous image dataset, in which each image is classified into one of 20 categories. Since we focus on statistical learning rather than computer vision, extracting features from images is out of our concern. Therefore, we will provide the pre-extracted features from the dataset instead of the original images.

There are totally 6666 image features for training, and we have another 2857 features are for testing. You may split a validation dataset with a prefered ratio by yourself. The features are all 1536 dim vectors, stored in numpy format with their labels. You may simply load it with `numpy.load` function. However, simply classifying such features is simple and well studied. To make our project more realistic and challenging, **while the training data is kept clean, the test dataset is noisy, which means that we have added random noises to these features. This indicates that the robustness of your model might be the key to test accuracy.**

Objective: train statistical learning models with data we provided, and achieve as high test accuracy as you can. Detailed descriptions are listed below.

## 2 Requirements

In this project, you should:

- try at least **three** different statistical learning methods, and at least **two** of them are **NOT** deep learning. We recommand methods learnt in this course.
- use at least **two** model assessment and selection methods to choose the best model and enhance the robustness and generalization ability of your models.
- submit a report, your codes, and the results on test dataset following the instructions in section 3.

Please be awear that though test accuracy (with its ranking) forms part of your score, it is not the only one. The detailed analysis, extensive experiments, a clear and well-written report, mannual implementation of the algorithms based on basic libraries (*e.g.*, `numpy`), reasonable modifications to standard algorithms and other highlights all contributes to a good project with high score. You may not wish to be stuck in the minor improvements of performance and ranking.

## 3 Submissions

You should submit your report and code via Canvas, and submit your results on test dataset with Kaggle platform. The report and code should be packaged to a zipped file, named with your student ID and name, with structure like:

- ZhangSan-022XXXXXXXXX.zip
  |_ Report.pdf

|_ Code
  |_ Readme file
  |_ Your code files here...

### 3.1 Report

Your report should describe how you complete this project, including:

- data processing,
- introduction of used models,
- model assessment and model selection methods, and how do you deal with the noise,
- the evaluation of your models,
- the conclusion,
- any other things you want to report in this project.

The report should be submitted as **a PDF file**, and recommend to have **six** or more pages. We suggest to use the LaTeX templates provided by top conferences and journals.

### 3.2 Code

Your code will be used for checking reproducibility. Therefore, your code should contain a `Readme` file, describing how to reproduce the results you submitted in kaggle and mentioned in the report. It coule be a `txt` file for simple cases and recommended to be a `PDF` file if its complex.

### 3.3 Test Results

Kaggle Link: https://www.kaggle.com/t/b4643612b511b82e3a0e3ba27d17706b

You need to submit your test results in this kaggle competition. You should register before the registration deadline and submit your results before the submission deadline. You are also required to include the screen shoots of each method's accuracy in your report.

## 4 Deadline

Kaggle Registration Deadline: December 10, 2022

Kaggle & Canvas Submission Deadline: January 8, 2023.