# COMP5328 - Advanced Machine Learning

Assignment 2

Due: 19 November 2020, 23:59PM

This assignment is to be completed in groups of 2 to 3 students. It is worth 25% of your total mark.

## Introduction

The objective of this assignment is to build an transition matrix estimator and two classification algorithms that are robust to label noise.

Three input datasets are given. For each dataset, the training and validation data contains class-conditional random label noise, whereas the test data is clean. You need to build **at least two different classifiers** trained and validated on the noisy data, that can have a good classification accuracy on the clean test data. You are required to compare the robustness of the two algorithms to label noise.

For the first two datasets, the transition matrices are provided. You can directly use the given transition matrices for designing classifiers that are robust to label noise.

For the last dataset, the transition matrix is not provided. You are required to build **an transition matrix estimator** to estimate the transition matrix. Then, employ your estimated transition matrix for classification. Your estimated transition matrix must be included in your final report. Note that to validate the effectiveness of your transition matrix estimator, you could use your estimator on the first two datasets and compare your estimation to the given transition matrices. The code contained in tutorial 9 could be a good starting point.

Data prepossessing is allowed, but please remember to clarify and justify it in the report carefully.

# 1 A Guide to Using the Datasets

Three image datasets with .npz format are provided. You can download them via canvas.

## 1.1 Attributes Contained in a Dataset

The following code is used to load a dataset and check the shape of its attributes.

```
import numpy as np
# Remember to replace the $FILE_PATH
dataset = np.load($FILE_PATH)
Xtr_val = dataset['Xtr']
Str_val = dataset['Str']
Xts = dataset['Xts']
Yts = dataset['Yts']
print(Xtr_val.shape)
print(Str_val.shape)
print(Xts.shape)
print(Yts.shape)
```

### 1.1.1 Training and validation data

The variable **Xtr_val** contains the **features** of the training and validation data. The shape is $(n, image\_shape)$ where $n$ represents the total number of the instances.

The variable **Str_val** contains the **noisy labels** of the $n$ instances. The shape is $(n, )$. For all datasets, the class set of the noisy labels is $\{0, 1, 2\}$.

**Note that** do not use all the $n$ examples to train your models. You are required to independently and randomly sample **80%** of the $n$ examples to train a model and use the rest **20%** examples to validate the model.

### 1.1.2 Test data

The variable **Xts** contains **features** of the test data. The shape is $(m, image\_shape)$, where $m$ represents the total number of the test instances.

The variable **Yts** contains the **clean labels** of the $m$ instances. The class set of the clean labels is also $\{0, 1, 2\}$.

## 1.2 Dateset Description

### 1.2.1 FashionMINIST0.5.npz

Number of the training and validation examples $n = 18000$.

Number of the test examples $m = 3000$.

The shape of each example $image\_shape = (28 \times 28)$.

The transition matrix $T = \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix}$.

### 1.2.2 FashionMINIST0.6.npz

Number of the training and validation examples $n = 18000$.

Number of the test examples $m = 3000$.

The shape of each example $image\_shape = (28 \times 28)$.

The transition matrix $T = \begin{bmatrix} 0.4 & 0.3 & 0.3 \\ 0.3 & 0.4 & 0.3 \\ 0.3 & 0.3 & 0.4 \end{bmatrix}$.

### 1.2.3 CIFAR.npz

Number of the training and validation examples $n = 15000$.

Number of the test examples $m = 3000$.

The shape of each example $image\_shape = (32 \times 32 \times 3)$.

The transition matrix $T$ is unknown.

## 2 Performance Evaluation

The performance of each classifier will be evaluated with the top-1 accuracy metric, that is,

$$\text{top-1 accuracy} = \frac{\text{number of correctly classified examples}}{\text{total number of test examples}} * 100\%.$$

To have a rigorous performance evaluation, you need to **train each classifier at least 10 times with the different training and validation sets generated by random sampling**. Then report both the **mean** and the **standard derivation** of the test accuracy.

## 3 Tasks

You need to implement at least **two label noise robustness classifiers** with at least one not taught in this course, and test their performance on the three datasets. You need to implement **an estimator to estimate the transition matrix**. The code must be written in Python 3. You are allowed to use external libraries for optimisation and linear algebraic calculation. If you have any ambiguity whether you can use a particular library or a function, please post your question on canvas or Ed.

### 3.1 Image Classification with Known Flip Rates

For the first two datasets, the transition matrices are provided. You can directly use the given transition matrices for designing classifiers that are robust to label noise. As mentioned in the **section 2**, for each classifier, you should report the mean and the standard derivation of the test accuracy.

### 3.2 Image Classification with Unknown Flip Rates

For the last dataset, Since the transition matrix is not provided, you need to implement an estimator to **estimate the transition matrix**. Then use the estimated transition matrix to build a noise robust classifier. Note that you can use the provided transition matrices of the first two datasets to validate the effectiveness of your transition matrix estimator. You need to include your estimated transition matrix in the final report. You also need to report the mean and the standard derivation of the test accuracy for each of your designed noise robustness classifier. Both estimation accuracy of the transition matrix and the test accuracy on the last dataset contribute to the final mark.

### 3.3 Report

The report should be organized similar to research papers, and should contain the following sections:

- In **abstract**, you should briefly introduce the topic of this assignment, your methods, and describe the organization of your report.
- In **introduction**, you should first introduce the problem of learning with label noise, and then its significance and applications. You should give an overview of the methods you want to use.
- In **related work**, you are expected to review the main idea of related label noise methods (including their advantages and disadvantages).
- In **methods**, you should describe the details of your classification models, including the formulation of the cost functions, the theoretical foundations or views (if any) of the cost functions, and the optimization methods. You should describe the details of the transition matrix estimation methods, theoretical foundations (if any), and optimization algorithms.
- In **experiments**, you should introduce your experimental setup (e.g., datasets, algorithms, evaluation metric, etc.). Then, you should show the experimental results, compare, and analyze your results. If possible, give your personal reflection or thoughts on these results.
- In **conclusion**, you should summarize your methods, results, and your insights for the future work.
- In **references**, you should list all references cited in your report and formatted all references in a consistent way.
- In **appendix**, you should provide instructions on how to run your code.

# 4 Submission guidelines

1. Go to Canvas and upload the following files/folders compressed together as a zip file.
   (a) report (a pdf file)
   The report should include all member's details (student IDs and names).
   (b) code (a folder)

i. algorithm (a sub-folder)
   Your code (could be multiple files or a project)

ii. input data (a sub-folder)
   Empty
   Please do NOT include the dataset in the zip file as they are large.
   We will copy the dataset to the input folder when we test the code.

Only one student needs to submit the zip file which must be named as student ID numbers of all group members separated by underscores. E.g. "xxxxxxxx_xxxxxxxx_xxxxxxxx.zip".

2. A plagiarism checker will be used.

3. A penalty of minus 5% marks per each day after the due date (email late submissions to the teaching assistant and confirm late submission dates with him). Maximum delay is 5 (five) days, after that assignments will not be accepted.

4. Remember, the submission deadline is **19 November 2020, 23:59PM**.

# 5 Marking scheme

| Category | Criterion | Marks | Comments |
|---|---|---|---|
| Report [80] | Abstract [3]<br>•problem, methods, and organization<br><br>Introduction [6]<br>•the problem you intend to solve<br>•the importance of the problem<br><br>Previous work [8]<br>•previous relevant methods used in literature<br>•their advantages and disadvantages<br><br>Label noise methods with known flip rates [23]<br>•pre-processing (if any)<br>•label noise methods' formulation<br>•cross-validation method for model selection or avoiding overfitting (if any)<br>•experiments<br>•discussions<br><br>Noise rate estimation method [12]<br>•noise rate estimation method's formulation<br>•experiments<br>•discussions<br><br>Label noise methods with unknown flip rates [10]<br>•pre-processing (if any)<br>•label noise methods' formulation (if different from above)<br>•cross-validation method for model selection or avoiding overfitting (if any)<br>•experiments<br>•discussions<br><br>Conclusions and future work [3]<br>•meaningful conclusions based on the results<br>•meaningful future work suggested | | |

| | | | |
|---|---|---|---|
| | Presentation [8]<br>•academic style, grammatical sentences, no spelling mistakes<br>•good structure and layout, consistent formatting<br>•appropriate citation and referencing<br>•use graphs and tables to summarize data<br><br>Other [7]<br>•at the discretion of the assessor: illustrate outstanding comprehensive theoretical analysis, demonstrate the insightful and comprehensive assessment of the significance of their results, provide descriptions and explanations that have depth but clarity, and are concisely worded | | |
| Code [20] | •reasonable code running time<br>•well organized, commented and documented | | |

Note: Marks for each category is indicated in square brackets. The minimum mark for the assignment will be 0 (zero).