Applied Machine Learning Systems ELEC0134 (23/24)

Assignment

General Overview

The AMLS assignment comprises individual code writing, training and testing on data, and an individual report in the form of a conference paper and (optionally) supplementary material. You are allowed to discuss ideas with peers, but your code, experiments, and report must be done solely based on your own work.

Assignment summary

1. The assignment leverages elements covered in:
    a. The AMLS lectures,
    b. The AMLS lab sessions, and
    c. Relevant research literature associated with machine learning systems.

    The assignment involves the realisation of various machine learning tasks on provided datasets. You are expected to go through the data, analyse it and/or pre-process it as necessary.
2. Using your ML knowledge acquired in the lectures and the labs, design solutions for each task described in the section *Assignment Description* below. You should also search the relevant literature for additional information, e.g., papers on state-of-the-art methods in machine learning.
3. Implement your solution in your preferred programming language, e.g., MATLAB, Python, C/C++, Java, etc. However, please note that the weekly exercise of the module will be based on Python, so you are encouraged to use this programming language too.
4. Write a report summarising all steps taken to solve the tasks, explaining your model and design choices. In addition, in the report, you should also describe and analyse the results obtained via your experiments and provide accuracy prediction scores on unseen data. Please refer to Report and Code Format and Marking Criteria section for more details about the report.

Goal of the assignment

- To further develop your skills and understanding of machine learning systems.
- To further develop your programming skills.
- To acquire experience in dealing with real-world data.
- To develop good practice in model training, validation and testing.
- To read state-of-the-art research papers on machine learning systems and understand the current challenges and limitations.
- To develop your writing skills by presenting your solutions and findings in the form of a conference paper.

Assignment Description

Datasets

We are going to use two datasets which have been chosen specifically for this assignment, as follows:

1. PneumoniaMNIST (https://medmnist.com/). The "PneumoniaMNIST" dataset comprises grayscale chest X-ray images, each with a consistent resolution of 28x28 pixels, mimicking the format of the classic MNIST dataset. These images are specifically categorized with binary labels indicating "Normal" (no pneumonia) or "Pneumonia" (presence of pneumonia). The dataset is tailored for binary classification tasks in machine learning, focusing on the automated diagnosis of pneumonia from standardized medical imagery. (Yang, J., Shi, R., Wei, D. et al. MedMNIST v2 - A large-scale lightweight benchmark for 2D and 3D biomedical image classification. Sci Data 10, 41 (2023).)
2. PathMNIST (https://medmnist.com/). The PathMNIST dataset is created based on previous research that aimed to predict survival from colorectal cancer histology slides. It provides a dataset (NCT-CRC-HE-100K) with 100,000 non-overlapping image patches from hematoxylin & eosin stained histological images. Additionally, there's a test dataset (CRC-VAL-HE-7K) containing 7,180 image patches from a different clinical center. This dataset includes images of nine different types of tissues, making it suitable for a multi-class classification task. To make the data more manageable, the original images, which were initially $3 \times 224 \times 224$ in size, have been resized to $3 \times 28 \times 28$. The NCT-CRC-HE-100K dataset is split into training and validation sets using a 9:1 ratio, while CRC-VAL-HE-7K serves as the test set for evaluation. (Yang, J., Shi, R., Wei, D. et al. MedMNIST v2 - A large-scale lightweight benchmark for 2D and 3D biomedical image classification. Sci Data 10, 41 (2023).)

The datasets you are going to use in this assignment are:

1. PneumoniaMNIST. This dataset contains 4,708 / 524 / 624 images for Training / Validation / Test. It is going to be used for task A.
2. PathMNIST. This dataset contains 89,996 / 10,004 / 7,180 images for Training / Validation / Test. It is going to be used for task B.

The datasets can be downloaded via following link:

https://medmnist.com/

Please obtain the training, validation, and test datasets in accordance with the original dataset. Feel free to use a portion of the training and validation datasets for model training. However, it is important to utilize the entire test dataset for conducting your testing and include the testing results in your report. You are welcome to use the machine learning model and feature extractor, if applicable, that you deem appropriate. The choice of models ranges from basic models such as SVM and Random Forest to neural networks, and more, depending on your preferences.

Tasks

The machine learning tasks include:

> **A**: Binary classification task (using PneumoniaMNIST dataset). The objective is to classify an image onto "Normal" (no pneumonia) or "Pneumonia" (presence of pneumonia)
>
> **B**: Multi-class classification task (using PathMNIST dataset): The objective is to classify an image onto 9 different types of tissues.

You should design separate models for each task, report training, validation, and testing errors / accuracies, along with describe any hyper-parameter tunice process. You are allowed to use the same model/methodology for different tasks, but you must explain the reason behind your choices. If you tried several models for one task, feel free to show them in your code and compare the results in the report.

Report and Code Format, and Marking Criteria

Report format and template

We provide both latex and MS word templates in ***AMLS_assignment_kit*** (https://bit.ly/AMLS_I_assignment_kit). The criteria for each part are detailed in the template. For beginners in latex, we recommend overleaf.com, which is a free online latex editor.

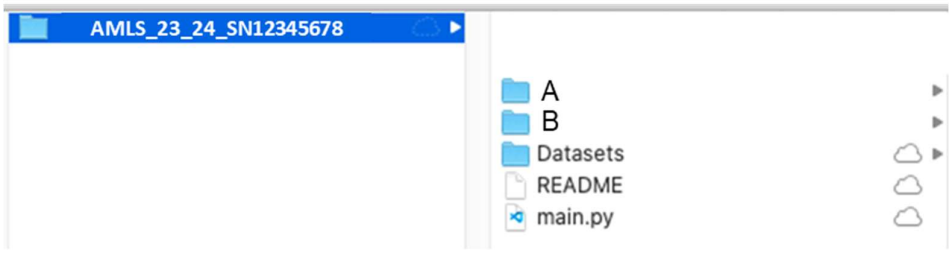
Your report should be no longer than **8 pages** (including the reference). You are allowed to append additional supplementary material to your report of up to additional **4 pages**.

Once you finish your report, please export it into a PDF document and name it with the following format (Using your SN number):

Report_AMLS_23-24 _SN12345678.pdf

Code criteria

You should write your code in modules and organize them in the following fashion:



- Keep **'Dataset'** folder empty while submitting your code. Use this folder for your programming assignment only. If you need to pre-process the dataset, do not save the

intermediate results or pre-processed dataset. Your final submission must directly read the files we provided.

- When assessing your code, we will copy-paste the dataset into this folder. Then, your project should look like:
    - AMLS_23-24_SN12345678
        - A
        - B
        - Datasets
            - PneumoniaMNIST
            - PathMNIST
        - main.py
        - README.md
- The '**A**', and '**B**' folders should contain the code files for each task.
- Pre-trained models (especially for deep learning models) are allowed to be saved in the folder for each task.
- The **README** file should contain:
    - a brief description of the organization of your project;
    - the role of each file;
    - the packages required to run your code (e.g. numpy, scipy, etc.).

  The recommended format for **README** file is markdown (**.md**). **.txt** is acceptable too.
- We should be able to run your project via '**main.py**'.
- You are NOT going to upload your code and dataset to Moodle. Please refer to Submission section for more details.

Marking scheme

The mark will be decided based on both the **report** and **corresponding code**. In particular, we will mark based on following scheme:

REPORT		**80%**	**CORRESPONDING CODE**	**20%**
Abstract		5%		
Introduction		5%		
Literature survey		5%		
Description of models (Use flow charts, figures, equations etc. to explain your models and justify your choices)	Task A	10%		
	Task B	10%		
Implementation (the details of your implementation, explain key modules in your code.)	Task A	10%	Correct implementation	10%
	Task B	10%	Correct implementation	10%

Experimental Results and Analysis	Task A	10%		
	Task B	10%		
Conclusion		5%		

It should be noted that – whereas we expect students to develop machine learning models delivering reasonable performance on tasks A and B – the assessment will not be based on the exact performance of the models. Instead, the assessment will predominantly concentrate on how you articulate about the choice of models, how you develop/train/validate these models, and how you report/discuss/analyse the results.

Submission

- **Deadline**: Please see the ELEC0134 AMLS I Moodle page.
- **Report submission**: you should only submit your report on Moodle:
- **Code submission**: You must include a link to your code in a repository that is publicly accessible in your report, but the link is hidden (e.g., GitHub, public Dropbox or Google Drive link, or similar).

  You are encouraged to use GitHub to save and track your project as we expect to see you progress your assignment gradually. Use your UCL GitHub account (or create an account) to start a git repository named

  AMLS_assignment23_24/

  Make sure to back-up your code on the git repository regularly and keep your repository private so it is not viewable by other students. Changes made after the assignment deadline will not be taken into account. The code should be well documented (i.e., each class and function should be commented) and an additional README.md file containing instructions on how to compile and use your code should be created in the repository. We reserve the right to test the code and we may ask you to provide us with your GitHub commit history evidencing how you gradually built and tested your solution.