Applied Machine Learning Systems ELEC0134 (21/22)

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 a provided dataset. 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 programming skills.
- To further develop your skills and understanding of machine learning systems.
- 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

Dataset

We provide one dataset which is designed specifically for this assignment and contains pre-processed subsets from the following dataset:

1. **Brain Tumor Classification (MRI) dataset:** an image-based dataset containing MRI scans of human brain. The images are in grayscale and are manually classified into 4 classes based on tumor type.

   Citation:

   Sartaj Bhuvaji, Ankita Kadam, Prajakta Bhumkar, Sameer Dedge, and Swati Kanchan, "Brain Tumor Classification (MRI)." Kaggle, 2020, doi: 10.34740/KAGGLE/DSV/1183165.

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

1. **AMLS-2021_dataset.zip:** 3000 512x512 pixel gray-scale MRI images organized in 4 classes.
2. **AMLS-2021_test.zip:** A separated test set with 200 images. This dataset will be released one week before the deadline. This test set is to help you to check whether your models are overfitted. It is optional to test your models on this dataset.

Download:

1. **AMLS-2021_dataset.zip:** http://shorturl.at/hquDP
2. **AMLS-2021_test.zip:** Please see ELEC0134 AMLS I Moodle page (available one week prior deadline).

Tasks

The machine learning tasks include:

A. **Binary task**

   Build a classifier to identify whether there is a tumor in the MRI images.

B. **Multiclass task**

   Build a classifier to identify the type of tumor in each MRI image (meningioma tumor, glioma tumor, pituitary tumor or no tumor).

You should design separate models for each task and report training errors, validation errors, and hyper-parameter tuning procedures. You are allowed to use the same model/methodology for different tasks, but you must explain the reason behind your own choices. If you tried several models for one task, feel free to show them in your code and compare the results in the report. Among all the models you may try for both tasks, you should implement at least one non-deep learning model (e.g. For task A: one SVM model and one deep learning model; for task B: one deep learning model).

Report and Code Format, and Marking Criteria

Report format and template

We provide both latex and MS word templates in ***AMLS_assignment_kit*** (available on ELEC0134 AMLS I Moodle page). 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 an additional supplementation material to your report with no longer than **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_21-22 _SN12345678.pdf

Code criteria

Gradual work:

- All of you are asked to use GitHub to save and track your project as we expect to see you progress in your assignment gradually over the term. 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.
- Your GitHub repository should NOT contain your report.

Submission package:

- The submission package refers to the code and README file of this assignment. You could also include the dataset, pre-processed datasets, saved models or any other necessary files to run your code in the package. The whole package should not exceed 1GB.
- We should be able to use your submission package to:
    - Reproduce the results in your report.
    - Reproduce the training process of your models.
- you can use cloud drives (e.g. Google Drive) to submit your submission package. You can also make your GitHub repository public at the deadline to submit your work. Make sure you make your shared files/GitHub repository publicly available. Put the link to access your package in the footnote of your report.
- Name your submission package (folder, zip file or GitHub repository) in the following format (Using your SN number):

AMLS_21-22 _SN12345678

- Your submission package should contain a README file that describes:

- A brief introduction of the task.
- A brief description of the organization of the files.
- The role of each file.
- How to run your code.
- Necessary packages or header files (e.g. numpy, scipy, etc.).
- The recommend format for README file is markdown (.md).

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		60%	CORRESPONDING CODE	40%
Abstract		5%		
Introduction		7%		
Literature survey		15%		
Description of models (Use flow charts, figures, equations etc. to explain your models and justify your choices)	Task A	4%		
	Task B	4%		
Implementation (the details of your implementation, explain key modules in your code.)	Task A	3%	Correct implementation	14%
	Task B	3%	Correct implementation	14%
Experimental Results and Analysis.	Task A	8%	Reasonable results	6%
	Task B	8%	Reasonable results	6%
Conclusion		3%		

More details about the role of each part could be found in the report template provided.

It should be noted that – whereas we expect students to develop machine learning models delivering reasonable performance on tasks – 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**: Use cloud drives or GitHub (or both) to share your code. Put the link in the footnote of your report. See code criteria section for more details.
- Do not put your name anywhere on the work that you upload – only use your UCL student number.