AI in Medicine I
WiSe 2023/24

Technical University Munich
Department of Informatics

I31 - AI in Medicine and Healthcare
I32 - Computational Imaging in AI and Medicine

Practical exercise 1

26 Oct 2023

Chest and breast classification

Submission deadline: 10 Nov 2023, 11:59 p.m.

Please submit your solutions (via Moodle).
The corresponding tutorial session is

26 Oct 2023, 16:15-17:45 in lecture hall 5901. EG. 051

For questions regarding this exercise sheet, please contact: shuting.liu@tum.de or yundi.zhang@tum.de
For general questions, please contact: course.aim-lab@med.tum.de

Code and further instructions on this practical exercise can be found at TUM-Moodle. Regarding the submission, kindly organize all solutions in a Jupyter Notebook file. Please name the Jupyter Notebook using the **TUM-IDs of all group members** and ensure that all expected results can be displayed in one forward run over all cells.

Medical image classification is one of the most applicable downstream tasks in medical imaging analysis. In this tutorial, you will implement basic CNN models for 2-dimensional (2D) and 3-dimensional (3D) datasets from MedMNIST to become familiar with how to work on medical imaging classification.

1. (60%) Task 1: **2D breast ultrasound classification**

Let us start with a 2D breast ultrasound dataset-BreastMNIST.

(a) (10%) Dataset loading and visualization

Let's first get familiar with how to load datasets from MedMNIST and how to visualize them. The first step is to load dataset we need and encapsulate it into torch.utils.data.Dataloader form. Here you will have to fill the blanks in the data_loading part. You are expected to load the data correctly and visualize two samples with their shapes.

(b) (10%) Upsampling and downsampling

For better understanding on how upsampling and downsampling work in convolutional networks (CNNs), the goal here is to utilize upsampling and max_pool functions from torch.nn in the script.

(c) (20%) Model implementation

The goal here is to implement a classification model. A good practice is to start with a simple but powerful model like CNN. To this purpose, we provided a CNN structure waiting for you to implement. Please complete the __init__ method, the forward pass, the loss function, and optimizer.

(d) (20%) Training and evaluation

Now we will train our model for prediction. Please complete the train function. For evaluation, AUC and ACC are commonly used. Please complete the test function to print evaluation scores as well as the ROC curve. In addition, you are expected to tune hyper-parameters to make the model reach an average AUC of 0.8.

2. (30%) Task 2: **3D chest CT classification**

In the second task we will work on a 3D dataset-NoduleMNIST3D-to realize binary classification of the chest CT. Here you are expected to implement a 3D CNN to realize classification.

(a) (10%) 3D dataset loading

Similar to what you achieved in the first task, what you need to do is to make sure that the data loading process also works on 3D data. You are expected to show the details using the code we provided.

(b) (10%) 3D CNN Net implementation

The next step is to design a 3D CNN for this new dataset. As you are already familiar with how to implement a 2D CNN, this time please try to modify based on that.

(c) (10%) Training and evaluation

The goal here is to train your new mode. Please modify the train and test functions for task 1 if it is necessary. Note that train and test functions should work for both models. You are expected to tune the model you designed to achieve an average AUC over 0.8.

3. (10%) Task 3: **What are the strengths and weaknesses of classification models?**

In the previous tasks, you worked on CNN and 3D CNN. What are other networks that can realize image classification? (Please make at least 3 examples.) Could you discuss the pros and cons of CNNs and the networks you listed in medical image classification?