**Python exercises (50 points)**

In this exercise, we will load and plot some MRI data.

For this, you need to install the python package nibabel.

Then, import nibabel in python, along with os, numpy, and matplotlib.

For the first group of tasks, we will use a file with MRI data from the NI-EDU dataset. Download and unzip this file to your computer:

https://github.com/lukassnoek/NI-edu/archive/refs/heads/master.zip

Tasks:

1. Among the extracted files, find the file ‘anat.nii.gz’ and load it with the function nibabel.load(). From the loaded image, get both the image data and the header and save them to new variables. (check nibabel online documentation for help). What is the size of the image data in x, y, z dimensions? How many voxels does the image have and what is their size? (3 points)
2. From the image data, create the following 2D plots using the matplotlib function imshow():
   1. Sagittal view (slice along x axis) at approximately the middle. (5 points)
   2. Coronal view (slice along y axis) approximately the middle. (5 points)
   3. Axial view (slice along z axis) at approximately the middle. (5 points)

For full points, all plots should be in grey tones, show a colorbar, the lower part of the brain should be at the bottom of the image (in axial view, the posterior part), and the plot axes should be labeled.

*Tip*: Look up online documentation of numpy arrays if you have trouble getting the data in right shape, and look up online documentation of matplotlib and specifically the imshow() function in order to format the images correctly.

2d. Plot all these three plots next to each other (horizontally) in a single figure using matplotlib.pyplot.subplots() and title the subplots as ‘sagittal’, ‘coronal’, and ‘axial’. (4 points)

1. Add the following plots:
   1. Coronal view of the brain in which the cerebellum can be seen. (3 points)
   2. Axial view of the brain in which the cerebellum can be seen. (3 points)

3b. Is this a T1- or T2-weighted scan? Why do you think so? (3 bonus points)

Now, we’ll look at a simulated MRI dataset which is meant to show a brain with lesions as they occur in Alzheimer’s disease.

Using numpy, load the data from the file ‘sim\_brain.npy’ (on Moodle). This only has the image data, no metadata.

1. With fig, ax = plt.subplots(), create figure and axis objects (with a single subplot).

Create an axial plot of the data at z-index 100. With the patches.rectangle() function from matplotlib, draw red rectangles around the major brain areas that look like they contain lesions. (8 points)

*Tip*: If you have problems figuring out how to use certain matplotlib functions, you can almost always find some simple examples with Google search.

Finally, we’ll load a fMRI dataset, in which brain activity (or specifically, the BOLD signal) has been recorded at different time points.

1. Find the file ‘func.nii.gz’ from the extracted folder (see task 1) and load it in the same way as the anatomical dataset. Get the dimensions and units of the data and describe them. (3 points)
2. Create a figure with subplots and plot the data, axial view sliced at the middle, at 8 evenly-spaced time points, starting from t=0. Here, do not use grey tones, but a multi-color colormap. All plots should have the same mapping of colors to specific values (8 points).

6b. Format the plot so that the axis labels are not shown, but each subplot shows its time stamp as title. Choose a size for the whole figure so that the titles are not too small or too large. In the subplots, zoom in on the brain so that there is only little ‘empty space’ around it. (5 bonus points)

1. From the data slice you plotted for the first subplot (t=0), identify the voxel with the highest value. From the whole dataset, extract the time series for this voxel and plot it as a function of t. Label your x-axis correctly and correctly mark time points as x-axis ticks. (4 points)