CS463/516 Medical Imaging

Assignment 3 – BOLD fMRI *Due Monday July 6th at 11:55 pm*

Assignment is to be done individually or in pairs (2 people). Groups of 3 or more are not permitted.

Submission format: one partner should submit a .zip file to moodle with:

1) pdf containing images and descriptions showing you have completed all parts of the assignment/bonuses
    a. should also include most important code segments in pdf, so I can easily evaluate your algorithms
2) full python source code **in .py format**
3) A README file describing any incomplete parts of the assignment and your group member names.

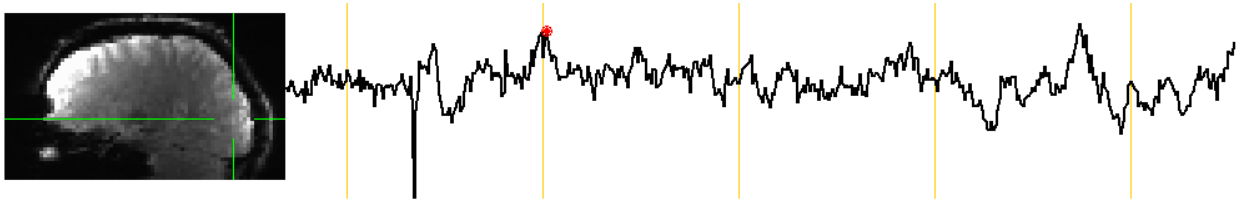
Motivation: Blood oxygen level dependent functional magnetic resonance imaging (BOLD fMRI) is the most common method for measuring human brain activity non-invasively in-vivo. BOLD fMRI images are 4-dimensional, consisting of a time series of 3d volumes, acquired in quick succession (every 1 or 2 seconds) typically over a period of 8-15 minutes (Figure 1)

Watch lectures 10 and 11 on BOLD: (or read the slides)

https://ensemble.ubishops.ca/hapi/v1/contents/permalinks/Ep5j4A9D/view

https://ensemble.ubishops.ca/hapi/v1/contents/permalinks/g8LWb96B/view

Figure 1: BOLD fMRI



In this assignment, you will work with the *'multisubject, multimodal face processing dataset'* available at openneuro.org. This dataset involves presentation of images of faces to the subject while acquiring BOLD fMRI images of the subject's brain activity. Your job is to preprocess these scans and then, in python, localize the brain area that processes faces.

Part 0: if you haven't already, install both afni and FSL software packages on a Linux or Mac operating system. If you use Windows, install VirtualBox and then download Neurodebian from the following link: https://neuro.debian.net/ and open it in VirtualBox by selecting File->Import Appliance. Neurodebian is a Linux distribution of the Debian flavor.

Once you have a Linux distro (Neurodebian or other, or you are on Mac) up and running, install afni and FSL libraries. See the respective website for installation instructions. Once afni and FSL are installed, you will be able to run the preprocessing pipeline attached with this handout.

You may need to add the following lines to the ~/.bashrc file in Linux, so that the FSL/afni commands will run:

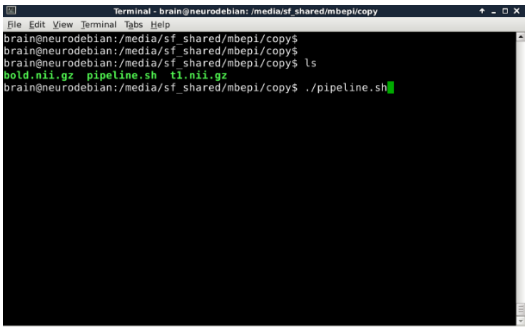
```
export PATH=$PATH:/usr/lib/afni/bin
export PATH=$PATH:/usr/local/fsl/bin
export FSLDIR=/usr/local/fsl ;
source $FSLDIR/etc/fslconf/fsl.sh
```

We will be working with data from openneuro in this assignment. link to sample dataset is below (subject-01). For the final part, you will use multiple subjects, and can download the data directly from openneuro using either the command line interface or from a browser.

Link to data + pipeline.sh:

https://drive.google.com/file/d/1rf_nnYXrOvm0-IUmf7D53bSKjP9jV3ej/view?usp=sharing

Part 1 - 25%: basic pre-processing: basic pre-processing (motion correction, bandpass filtering, spatial smoothing). The pre-processing pipeline 'pipeline.sh' has been provided in the google drive link containing the dataset. Download the dataset and run the pipeline from the command terminal. The pipeline will take roughly 5-10 minutes to run, depending on your hardware setup. The processing pipeline will produce a final output called 'clean_bold.nii.gz'. you will use this image for part 2 and 3.



In your pdf, show a screenshot of your directory after running 'pipeline.sh' and typing 'ls' at the command line.

Part 2 – 45%: localize task activation (to be done in python)
In lecture 10, task-based analysis is covered. Briefly, the procedure is as follows:

1) Clean the BOLD images (using pipeline.sh)
2) Load the clean_bold.nii.gz output by pipeline.sh image and events.tsv into your python environment
3) Using the timing from events, create an 'ideal time series' that represents how the brain should react to the stimulus (face=1, no face=0) – check lecture 10 slide 14 to see how this should look.
4) Convolve the ideal time series with the hemodynamic response function (HRF) provided in drive link
   Load hrf using:

```
import pandas as pd
hrf = pd.read_csv('c:/shared/hrf.csv',header=None)
```

5) Correlate the convolved ideal with the BOLD signal in each voxel.
6) Visualize the correlation map to see where in the brain the activation was strongest

2a: use the above steps to find the brain area which correlates to viewing of faces.

2b: **NO PREPROCESSING**: repeat the above, but leave out step 1 (use bold.nii.gz, with no pre-processing, instead of the clean_bold.nii.gz output by pipeline.sh)

In your pdf, for both 2a and 2b, display a figure with all z-slices (46 slices) from the final activation map using imshow with vim=-0.25 and vmax=0.25. There should be some clusters near the back of the brain with high correlation values. Comment on the difference between the correlation map produced with/without preprocessing. Are the correlations stronger or weaker when preprocessing is included? Why?

Part 3: - 30%: multi-subject analysis (combined python + afni, FSL). The dataset on openneuro contains scans from 16 subjects. In part 2, you were asked to only process data from a single subject, in fact, the google drive link contained data from sub-01. Now, you will download data from all 16 subjects (just the T1 and first fMRI, see below)

Each subject actually contains multiple BOLD fMRI runs, but you may use only the first. However, if you wish you may use all the BOLD scans from each subject and average the resulting correlation maps (to produce a clearer correlation).

batch processing: once you have each subjects' data downloaded, create a separate folder for each subject and place each subjects' data in their own separate folder. Then, run the pre-processing pipeline on each subject separately (this may take up to 1 hour). You can simply place the script in each subject's folder and run it as-is at the command line, or you can create a 'for' loop surrounding the preprocessing code in the .sh file to loop over all subjects automatically.

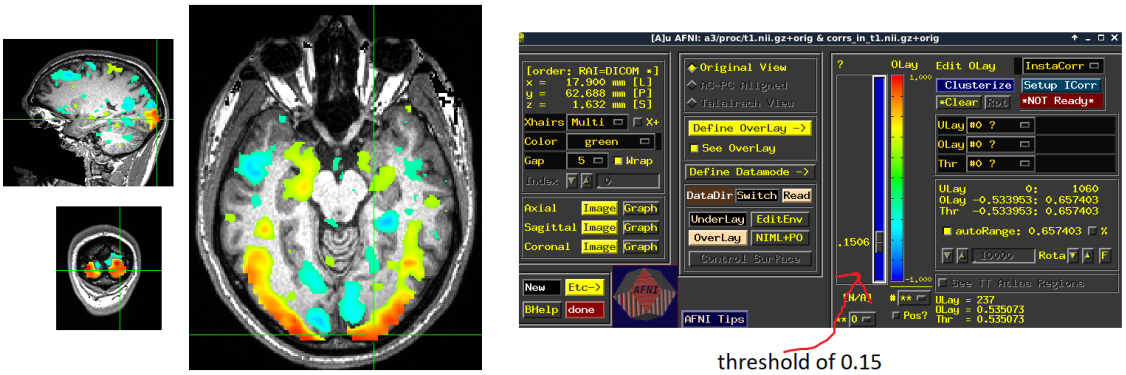
Once each subject has been pre-processed, run the correlation analysis as in part 2 on each subject and save the output as a .nii file into the same subject's directory. Be sure to keep the same header transform as the input image when saving using nibabel, otherwise the correlation map will not be in the same space (see below)

```
corrs_nifti = nib.Nifti1Image(corrs, fmri.affine)
# where fmri is the original nifti loaded by nibabel
nib.save(corrs_nifti,'c:/shared/a3/proc/corrs.nii.gz')
```

Using epi_reg.mat, bring the correlation map into the subject's T1 space (see below) and visualize the correlation map as an overlay on the T1 using afni.

```
brain@neurodebian:/media/sf_shared/a3/proc$ flirt -in corrs.nii.gz -ref t1.nii.gz -applyxfm -init epireg.mat -out corrs_in_t1.nii.gz
```

Increase the threshold to find the brain area of maximum activation, and in your pdf show each subject's correlation map overlayed on the T1 in afni with a threshold of 0.15:



e.g, your pdf should show a row of images (n=16) with each subject's correlation map overlayed on the T1 as above (axial view)

Group average: once you have obtained each subjects' correlation map in their native T1 space, the last step is to bring all subjects into the same space and average the correlation map, to create a 'grand average' of where the brain processes faces. You have been provided with a 'template brain' MNI152_2009_template.nii.gz in the drive link. Using flirt or ANTs, register each subject's skull-stripped T1 to this template image and save the transform. Then, apply the transform to the correlation map in T1 space, bringing each subjects' correlation map into template space. Finally, once all subjects' correlation maps have been aligned to the MNI152, average them, creating a 'grand average' correlation map. Display this map as an overlay on the MNI152 in afni, with a suitable threshold, showing the region of maximum correlation, and show this grand average correlation map overlayed on the MNI152 in the pdf.

Bonus 1 +6% - some faces may result in higher BOLD activation than others. Using events.tsv, and the knowledge that the hemodynamic response is roughly 4.5 seconds *after* the onset of the face, locate the area in the brain that distinguishes famous faces and unfamiliar faces. You can do this by isolating all BOLD time points 4.5 seconds after the onset of a famous face (F1) and also isolate all BOLD time points 4.5 seconds after onset of unfamiliar face (F2) and then do a t-test (F1-F2) which will then give a t-value and a p-value in each voxel showing which voxels had significant differences in their response from famous to unfamiliar faces. Show this grand average t-map overlayed on the MNI152.

Bonus 2 +10% - the dataset on openneuro also has MEG data. I don't really expect anyone to accomplish this, but if you're feeling ambitious, download a MEG dataset and using the *mne* software in python, apply source localization to locate where the evoked MEG response to the face is strongest. https://www.frontiersin.org/articles/10.3389/fnins.2013.00267/full