Assignment 3 Simultaneous EEG-fMRI
Due Wednesday July 7th at 11:59 PM

In this assignment you will get familiar with digital signal processing including common denoising techniques (ICA), filtering, resampling, and convolution. The goal of this assignment is to denoise an EEG dataset acquired during a simultaneous EEG-fMRI experiment, extract some frequency-domain measures from the denoised EEG signal, and correlate with fMRI (which is also a time series).

Background: simultaneous EEG-fMRI is a technique which combines EEG recording with fMRI acquisition, yielding two separate datasets, which when combined achieve excellent temporal resolution (due to the high temporal sampling rate of EEG) and excellent spatial resolution (due to the high spatial resolution of fMRI). Simultaneous EEG-fMRI allows to investigate the precise brain areas involved in the generation of frequency-specific brain oscillations. Here, we are interested in the alpha-band (8-13 Hz) of the EEG signal, and we want to find out which brain areas correlate with alpha band power. Therefore, we need to get the alpha-band power from the EEG data, and correlate it with BOLD signal in each voxel. To do so, there are some basic preprocessing steps (see below) that we need to perform.

Step 1: get the EEG and fMRI datasets (link). The EEG dataset contains 64 separate time series (one time series per electrode). The 32nd electrode (index 31) is an ECG (electrocardiogram) electrode which records from the heart, the other 63 electrodes record from the brain. The sampling rate of the EEG is 5000 Hz. The fMRI dataset is a 4D image (3D + time) with a temporal sampling rate of 1 Hz. An anatomical T1 is also included.

Step 2: denoise the EEG dataset. the EEG dataset was acquired simultaneously with the fMRI dataset, so it has two major artifacts 1) the gradient artifact (*figure 1*) and 2) the ballistocardiogram artifact (*figure 2*). The gradient artifact must be removed first, then the ballistocardiogram artifact can be removed second.

a) Load EEG dataset into python using mne
b) Remove gradient and ballistocardiogram artifacts (*figure 3*)
    a. Use average artifact subtraction for both (Allen et al. 2000 see moodle)
c) Run ICA *(figure 5)* on EEG signal to isolate the alpha band components (*figure 6)*
    a. can use the mne implementation of fastICA
d) Find the alpha band ICA components by visualizing the topography and power spectrum (*figure 7*).
e) Apply 8-13 Hz bandpass filter to alpha-band component(s) (to isolate alpha), rectify, and smooth.
    a. There could be multiple components with good alpha, in that case, average them (after rectifying)

Step 3: denoise the fMRI dataset. the fMRI dataset must be motion corrected and bandpass filtered:

a) Load fMRI dataset into python using nibabel
b) Motion correct by registering every volume to the first volume
    a. find some python library that can do this or create your own, can also do this using FSL or AFNI or some other external library (then load the corrected image in python)
c) Apply 0.01 – 0.1 Hz bandpass filter to the time series in each voxel
    a. This will remove all frequencies outside 0.01 – 0.1 Hz

Step 4: combine datasets. After completing the pre-processing for both datasets (step 2,3) it is time to combine the datasets and get our final result:

a) Resample the smooth, rectified 8-13 Hz bandpass filtered component (step 2e) to 1 Hz (to match fMRI).
b) Correlate EEG alpha power (8-13 Hz) with fMRI in each voxel, yielding a 106x106x16 image where the value in each voxel is a correlation coefficient.
    a. It might help to also bandpass filter the rectified, resampled EEG alpha between 0.01 – 0.1 Hz.
c) Show resulting correlation map overlayed on the T1 image (*figure 8*).
d) Perform cluster-based multiple comparison testing to eliminate spurious clusters

You must show your correlation map as overlay on the T1-weighted image, which means you will need to register the fMRI to the T1 first (check out *epireg* command from FSL libraries).

We will discuss the steps (average artifact subtraction, filtering, time-frequency analysis, etc.) in the next weeks (breakfast club and lectures.

Tips:

Step 2a: see code at end of this handout to get started loading the EEG data (need to install mne package).

Step 2b: To remove **gradient** artifact from EEG signal using average artifact subtraction, you need two things 1) the latency of the first artifact onset and 2) the duration of each individual artifact (repetition time (TR)). To get (1), use the R128 triggers from the EEG dataset. the latency of the first R128 trigger gives the onset of the first artifact. To get (2) we know that the fMRI has a sampling rate of 1 Hz, therefore, each gradient artifact is 5000 EEG samples in length (because this EEG dataset has a sampling rate of 5000 Hz). To remove **ballistocardiogram** artifact, you will need to isolate each individual heartbeat (*figure 4)* and then perform a similar procedure to gradient subtraction (get the average artifact and subtract it from each individual heartbeat instance, separately for each channel).

Step 2c: alpha band components are those with a 'smooth but concentrated' scalp topography like ICA015 in figure 5. If you inspect the power spectrum, you will see these components also have an alpha peak (more power from 8-13 Hz).

Step 4c: can use epi_reg (from FSL) to register the fMRI to the T1 (AFNI also has a function called 3dAllineate).

Step 4d: check out '3dclustsim' from AFNI for eliminating spurious correlation clusters (clusters which do not exceed some threshold for random noise).

Figure 1: Before gradient artifact subtraction

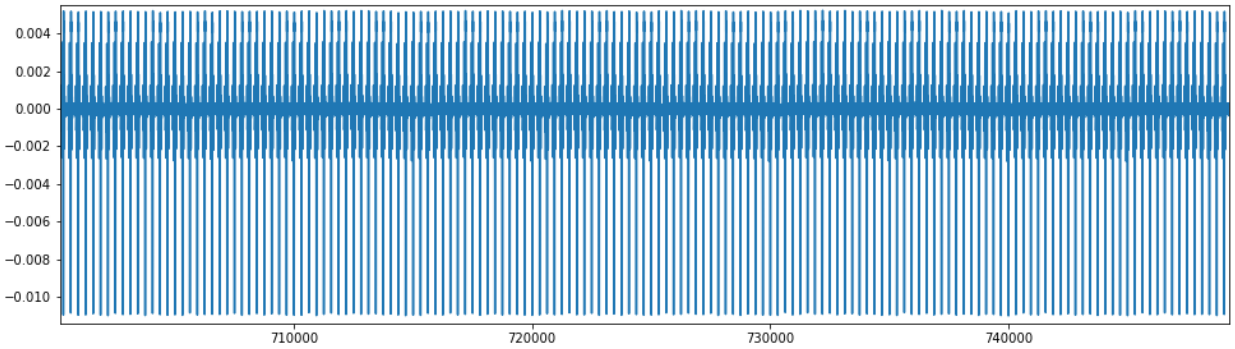


Figure 2: After gradient artifact subtraction but before ballistocardiogram artifact subtaction

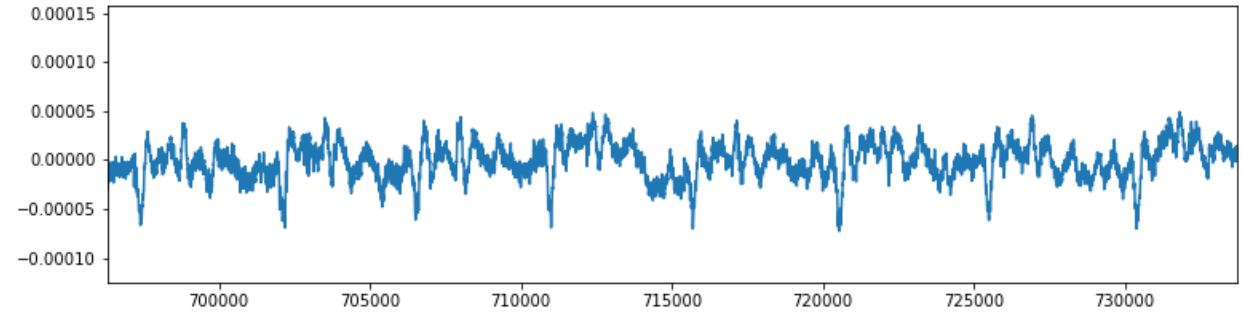


Figure 3: comparing *before* (blue) and *after* (orange) ballistocardiogram artifact subtraction

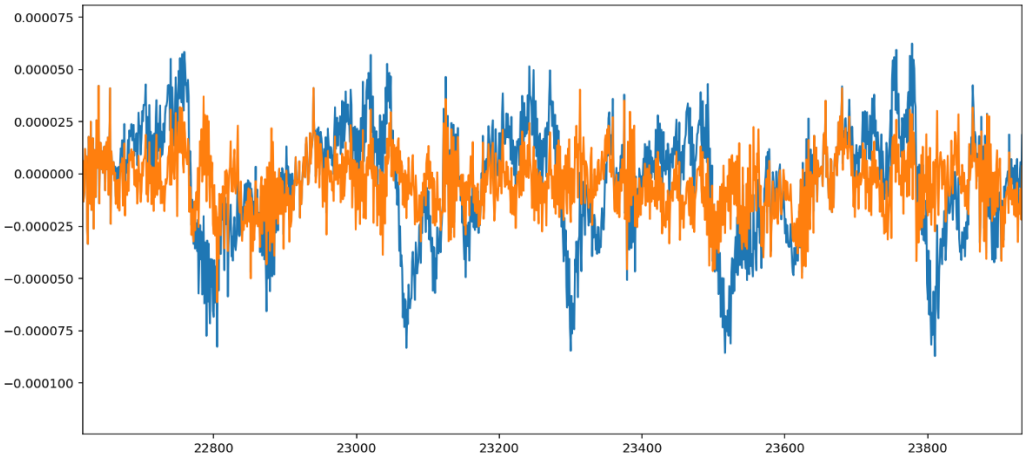


Figure 4: an example of detected ballistocardiogram artifact peaks (scipy.signal.find_peaks).

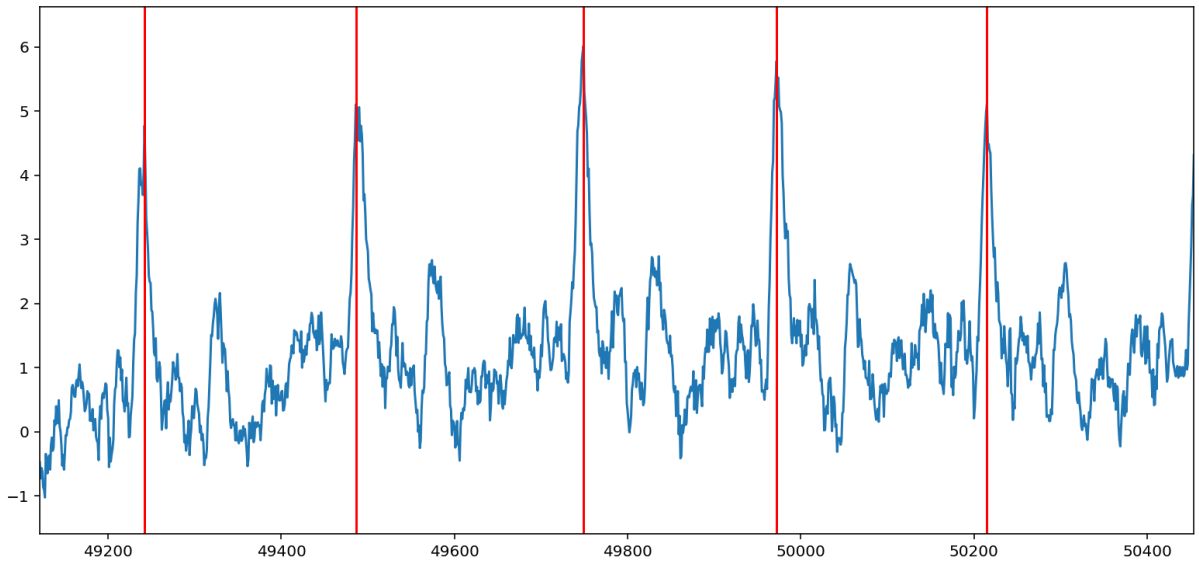


Figure 5: ICA component weight maps for components 0-19 (out of a total 63, only 0-19 shown here). Components 0,1 and 2 are definitely artifacts (Eye blinks and residual bcg). Components 5,15, and 19 are definitely neural in origin (what we want). The other components its more difficult to tell

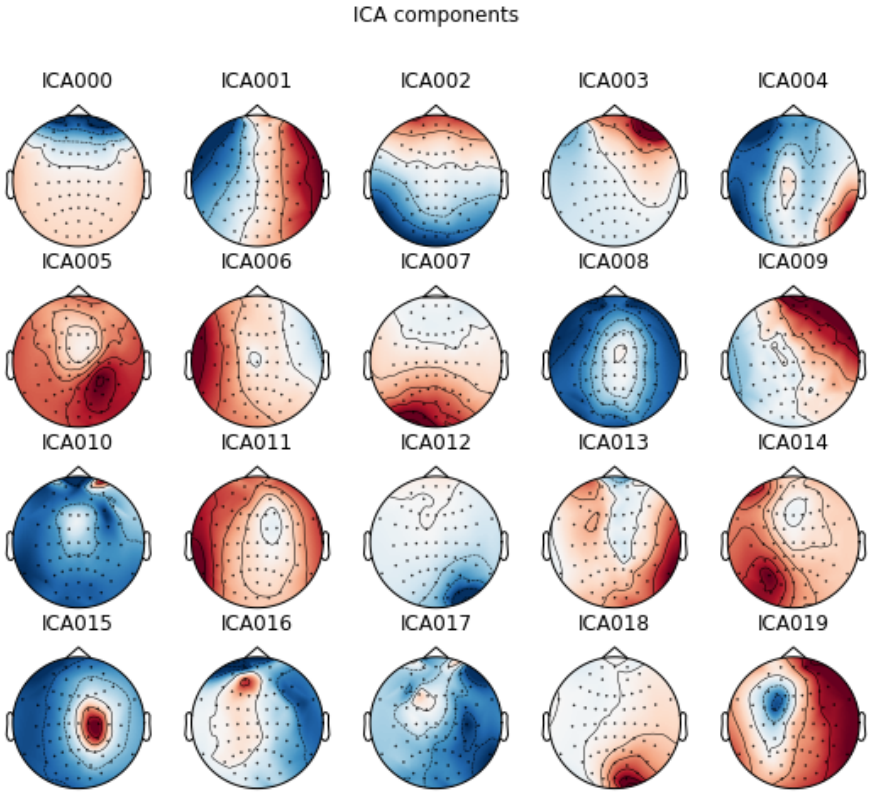


Figure 6: power (computed using scipy.signal.welch) for the time series of each component shown in figure 5. Note how components 5 and 15 (which are 'good' components, or neuronal components) have a strong peak in the alpha band (8-13 Hz). The peaks higher than 30 Hz are due to scanner noise (residual gradient artifact and helium pump artifacts).

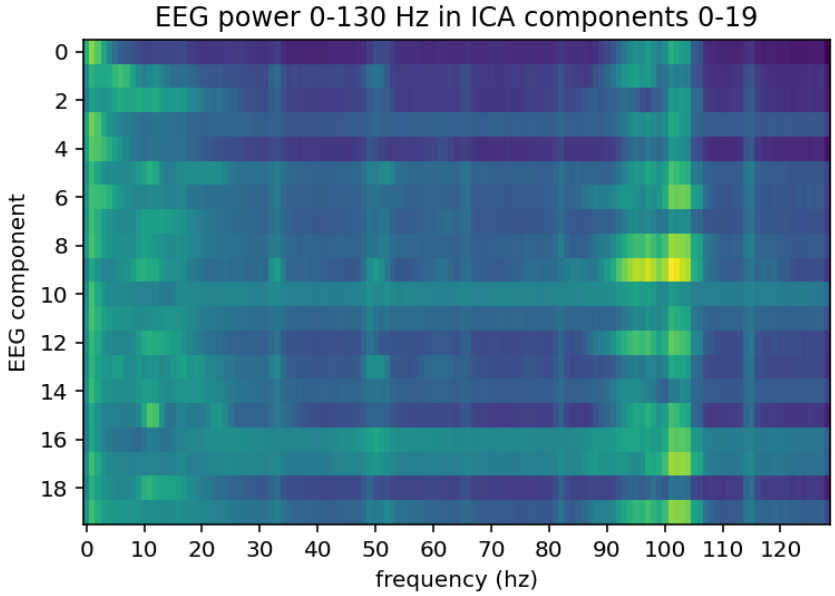


Figure 7: component 5 log power and scalp map (A good component). Even good components suffer from the residual peaks above 30 Hz because ICA fails to completely separate the artifacts.

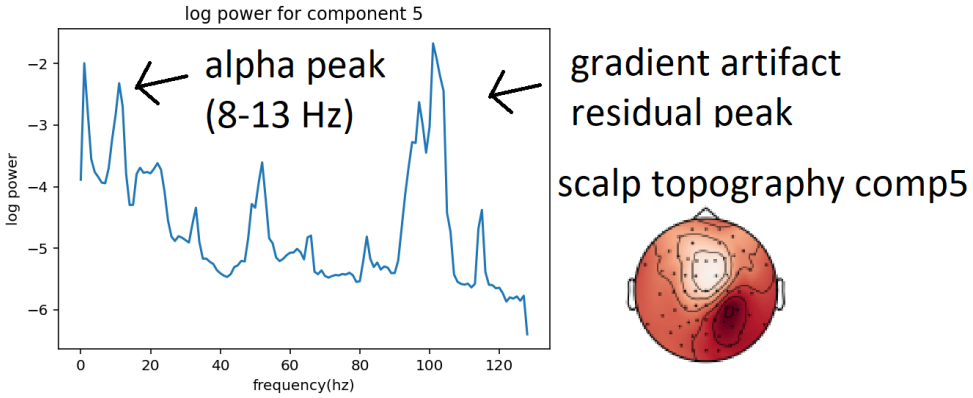
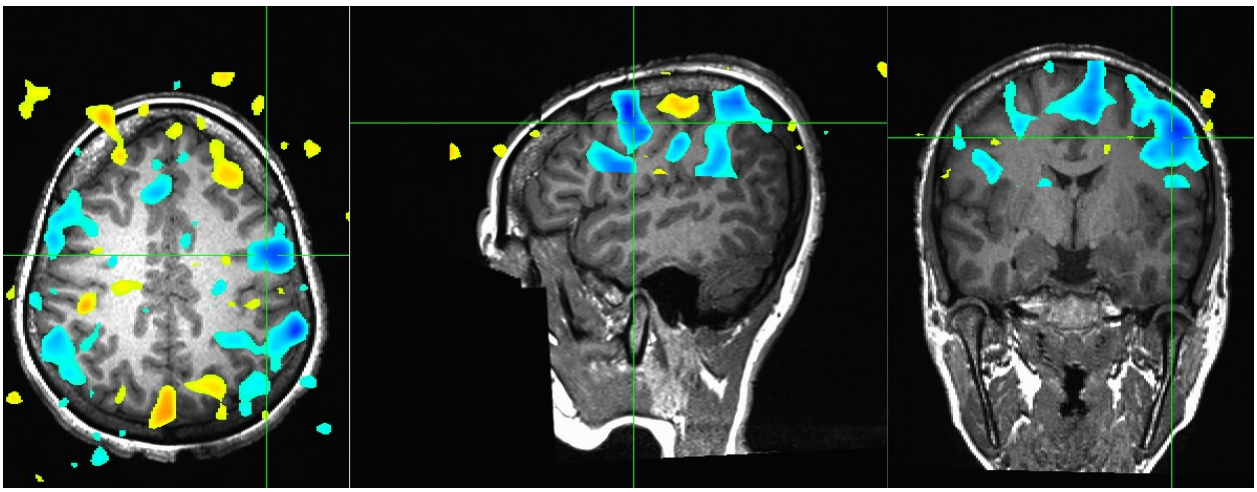


Figure 8: Final result. this is the result after correlating the resampled alpha band power with BOLD in each voxel and registering the correlation map to the T1. Note the inverse alpha vs fMRI correlation (blue) which is expected based on previous studies of alpha power and fMRI.



Starter script (to load the data)

```
import mne as mne
import nibabel as nib
import numpy as np
import matplotlib.pyplot as plt
from numpy.fft import fft, ifft, ifft2, fftfreq, fft2, fftshift,fftn, ifftn


eeg = mne.io.read_raw_brainvision('C:/shared/simeegfmri/subxp210/sub-xp210_task-2dNF_run-
02_eeg.vhdr',preload=True)
eegdata = eeg.get_data()

eeg.info['bads'] = ['ECG']
eeg.set_montage('standard_1020',raise_if_subset=False)

markers = mne.read_annotations('C:/shared/simeegfmri/subxp210/sub-xp210_task-2dNF_run-
02_eeg.vmrk')
nmrks = 324
r128s = np.zeros(nmrks)
count = 0
for i in markers:
    if i['description'] == 'Response/R128' and count < 324:
        r128s[count] = i['onset']
        count = count + 1
```

now you have the markers and can start doing the gradient artifact subtraction based on the R128 markers.

Dataset (from openneuro, if you choose to do the bonus):

A multi-modal human neuroimaging dataset for data integration: simultaneous EEG and fMRI acquisition during a motor imagery neurofeedback task: XP2

Bonus +10%: get ALL subjects and runs *(I only provided you with data for sub-xp210 run 02*) and process them as above. Create a 'grand average' correlation map by registering all subjects T1 to the MNI152 and using the matrix to bring each subjects' *individual* correlation map into the MNI space and then averaging all subjects (and runs) together. Show the grand average correlation map as overlay on the MNI152 anatomical T1 (this is equivalent to a 2-year master's research project, minus the actual data collection).