New analysis to determine the contribution of different visual areas in the EEG signal

This is different from source localization because we are not interested in the exact location of the source but how the response from different retinotopically defined areas will look on the topography.

Previously, Justin did the cruciform papers by aligning EGI with MRI scans and then ran a forward model to find the topographies created by different visual areas.

Now we are aligning the EGI with the Biosemi standard cap. The forward model is run on the EGI and sent onto the “fitted” Biosemi electrodes for each participant.

I think the final goal is to determine the contribution of different visual areas using a regression model to different EEG topography maps.

Might need stuff in svndl\_code/alesToolbox, fieldtrip-aleslab-fork/external/

Also needs the freesurfer preferences to be set properly:

Start terminal and run these commands.

*cd ~*

*nano .bash\_profile*

Now add the following line:

*export SUBJECTS\_DIR=/Volumes/4TB\_External/anatomy/FREESURFER\_SUBS/*

*Save:  ctrl-x  and say yes to save*

Command to run:

1. alignBioSemi128ToMrcElecV2(*fif file name of the participant*)
2. prepareInversesForMrc – this function used the preferences set in freesurfer.

Check getpref('freesurfer','SUBJECTS\_DIR') – if not correct see above

For checking that the inverses are fine:

ss=readDefaultSourceSpace('skeri0001')

fwd=mne\_read\_forward\_solution('skeri0001-fwd.fif')

sol = makeForwardMatrixFromMne(fwd,ss);

plotTopo(sol(:,1)) % visualize souce

(makeManyForwards, computeManyMrcInverses, makeInversesGui)

open skeriDefaultSimParameters

setpref('mrCurrent','AnatomyFolder','/Volumes/4TB\_External/anatomy')

addpath /Users/marleneponcet/Documents/Git/fieldtrip-aleslab-fork/external/mne

1. mrSimScript(projDir,params);

list of ROI:

*roiList = getRoisByType(roiDir,'func')*

Axx = load('Axx\_c999.mat');

Need to correct the Axx files

for i=1:length(fn);Axx.(lower(fn{i})) = Axx.(fn{i});end % lower capital for the field names

Axx.time = linspace(0,(Axx.nT-1)\*Axx.dTms,Axx.nT)

Axx.pval = zeros(128,108);

*Axx.amp = Axx.amp';*

*Axx.cos = Axx.cos';*

*Axx.sin = Axx.sin';*

*Axx.wave = Axx.wave';*

*Axx.confradius = Axx.pval;*

*Axx.freq = linspace(0,(Axx.nFr-1)\*Axx.dFHz,Axx.nFr)';*

cfg.layout = 'biosemi128.lay';

interactiveSteadyStatePlot2(cfg,Axx)