**Figure 6 – Figure supplement 1 – Source Code 1: Local field potentials analysis between the ventral hippocampus (vHip) and medial prefontal cortex (mPFC)**

clc

clear all

%Funcao adequada para os dados da Eduardo Campos (exp.LuísaPinto)

# Initialization of POI Libs

%USAR A FUNCAO XLWRITE EM VEZ DO XLSWRITE QUE SE USA NO WINDOWS

% Add Java POI Libs to matlab javapath

javaaddpath('poi\_library/poi-3.8-20120326.jar');

javaaddpath('poi\_library/poi-ooxml-3.8-20120326.jar');

javaaddpath('poi\_library/poi-ooxml-schemas-3.8-20120326.jar');

javaaddpath('poi\_library/xmlbeans-2.3.0.jar');

javaaddpath('poi\_library/dom4j-1.6.1.jar');

javaaddpath('poi\_library/stax-api-1.0.1.jar');

n\_group=2;

n=[10 8]; %no. files in each group

%Output file

output\_filename=strcat('LFP\_PP\_vHip\_novo\_',datestr(now,'yy-mm-dd\_HH-MM-SS'),'.xls');

% Line Filter (50 Hz)

hd=LineFilter;

% Multitaper parameters

tapers=[3 5];

pad=0;

Fs=1000; % FREQUENCIA DE SAMPLING

seg\_length=1\*Fs;

fpass=[0 100];

err= [2 0.05];

trialave=1;

%Rhythms

Delta=[0.5 4];

%Theta=[4 8];

Theta=[4 12];

Alpha=[8 13];

Beta=[12 20];

Gamma\_lo=[20 40];

Gamma\_hi=[40 100];

group=[];

coh=[];

psd\_ch1=[];

psd\_ch2=[];

spect\_ch1=[];

spect\_ch2=[];

CH1\_delta=[];CH1\_theta=[];CH1\_alpha=[];CH1\_beta=[];CH1\_gamma\_lo=[];CH1\_gamma\_hi=[];

CH2\_delta=[];CH2\_theta=[];CH2\_alpha=[];CH2\_beta=[];CH2\_gamma\_lo=[];CH2\_gamma\_hi=[];

CH1\_psd\_delta=[];CH1\_psd\_theta=[];CH1\_psd\_alpha=[];CH1\_psd\_beta=[];CH1\_psd\_gamma\_lo=[];CH1\_psd\_gamma\_hi=[];

CH2\_psd\_delta=[];CH2\_psd\_theta=[];CH2\_psd\_alpha=[];CH2\_psd\_beta=[];CH2\_psd\_gamma\_lo=[];CH2\_psd\_gamma\_hi=[];

f\_psd = Fs/2\*linspace(0,1,Fs/2+1);

Coh\_delta=[];Coh\_theta=[];Coh\_alpha=[];Coh\_beta=[];Coh\_gamma\_lo=[];Coh\_gamma\_hi=[];

FileName\_array={};

for g=1:n\_group

data=[];

eval(strcat('[FileName,PathName,FilterIndex]=uigetfile({''.txt''},''Group LFP ',num2str(g),' '',''Multiselect'',''on'');'))

%eval(strcat('[FileName\_art,PathName\_art]=uigetfile({''.mat''},''Group Artifact Marks ',num2str(g),' '',''Multiselect'',''on'');'))

for i=1:n(g)

file=strcat(PathName,FileName(i));

%file\_art=strcat(PathName\_art,FileName\_art(i));

FileName{i}

%FileName\_art{i}

FileName\_array=[FileName\_array FileName(i)];

[seg]=importdata(cell2mat(file));

%seg(:,1)=seg\_i(:,1)\*200;

%seg(:,2)=seg\_i(:,2)\*200;

%[art]=importdata(cell2mat(file\_art));

% eval(strcat('file=strcat(PathName,FileName(',num2str(i),'));'))

% eval('[seg]=importdata(file);')

% data(:,2) = mat2gray(data(:,2)) - mean(mat2gray(data(:,2)));

% data(:,3) = mat2gray(data(:,3)) - mean(mat2gray(data(:,3)));

% data=normc(data);

% eval('data=[data;seg];')

%Correct no. columns if necessary

%if (size(seg,2)>2)

% seg(:,1)=[];

%end

% Correct Sampling Rate if necessary

%if (size(seg,1)>25000)

% seg=downsample(seg,4);

%end

seg=filter(hd, seg);

n\_seg=size(seg,1)/seg\_length;

%exclude segments with artifacts

%markers\_ch1=ArtRej\_ExtremeValues(seg(:,1),th\_PFC(i,1),th\_PFC(i,2),200)';

%markers\_ch2=ArtRej\_ExtremeValues(seg(:,2),th\_HIP(i,1),th\_HIP(i,2),200)';

%

%art\_ch1=zeros(n\_seg,1);

%art\_ch2=zeros(n\_seg,1);

%for s=1:n\_seg

% seg\_m\_ch1=markers\_ch1((s-1)\*seg\_length+1:s\*seg\_length,1);

% seg\_m\_ch2=markers\_ch2((s-1)\*seg\_length+1:s\*seg\_length,1);

% if isempty(find(seg\_m\_ch1==1))==0

% art\_ch1(s)=1;

% end

% if isempty(find(seg\_m\_ch2==1))==0

% art\_ch2(s)=1;

% end

%end

%art\_m=art\_ch1+art\_ch2;

%seg\_art=[];

%sg=0;

%art\_m=zeros(1,100);

%for s=1:n\_seg

% sg=sg+1;

% seg\_art(:,sg)=art((sg-1)\*seg\_length+1:sg\*seg\_length-1,1);

% if isempty(find(seg\_art(:,sg)==1))==0

% art\_m(s)=1;

% end

%end

%Separate data in equally sized segments

%seg\_ch1=zeros(seg\_length,n\_seg);

%seg\_ch2=zeros(seg\_length,n\_seg);

seg\_ch1=[];

seg\_ch2=[];

sg=0;

for s=1:n\_seg

%if art\_m(s)==0

sg=sg+1;

seg\_ch1(:,sg)=seg((sg-1)\*seg\_length+1:sg\*seg\_length,1);

seg\_ch2(:,sg)=seg((sg-1)\*seg\_length+1:sg\*seg\_length,2);

%end

end

%n\_seg=sg;

%seg\_eliminados=100-sg;

%disp(seg\_eliminados)

params=struct('tapers',tapers,'pad',pad,'Fs',Fs,'fpass',fpass,'err',err,'trialave',trialave);

eval(strcat('[C,phi,S\_ch1xch2,S\_ch1,S\_ch2,f,confC,phistd,Cerr]=coherencyc(seg\_ch1,seg\_ch2,params);'))

%Power Spectral Density

%Pxx\_ch1=zeros(Fs,n\_seg);

%Pxx\_ch2=zeros(Fs,n\_seg);

Pxx\_ch1=[];

Pxx\_ch2=[];

for p=1:n\_seg

Pxx\_ch1(:,p)= 10\*log10( fft(seg\_ch1(:,p),Fs).\*conj(fft(seg\_ch1(:,p),Fs)) );

Pxx\_ch2(:,p)= 10\*log10( fft(seg\_ch2(:,p),Fs).\*conj(fft(seg\_ch2(:,p),Fs)) );

end

Mean\_Pxx\_ch1=mean(Pxx\_ch1,2);

Mean\_Pxx\_ch2=mean(Pxx\_ch2,2);

delta\_psd\_i = find(f\_psd >= Delta(1) & f\_psd < Delta(2));

theta\_psd\_i = find(f\_psd >= Theta(1) & f\_psd < Theta(2));

alpha\_psd\_i = find(f\_psd >= Alpha(1) & f\_psd < Alpha(2));

beta\_psd\_i = find(f\_psd >= Beta(1) & f\_psd < Beta(2));

gamma\_lo\_psd\_i = find(f\_psd >= Gamma\_lo(1) & f\_psd < Gamma\_lo(2));

gamma\_hi\_psd\_i = find(f\_psd >= Gamma\_hi(1) & f\_psd < Gamma\_hi(2));

CH1\_psd\_delta=[CH1\_psd\_delta; mean(Mean\_Pxx\_ch1(delta\_psd\_i))];

CH1\_psd\_theta=[CH1\_psd\_theta; mean(Mean\_Pxx\_ch1(theta\_psd\_i))];

CH1\_psd\_alpha=[CH1\_psd\_alpha; mean(Mean\_Pxx\_ch1(alpha\_psd\_i))];

CH1\_psd\_beta=[CH1\_psd\_beta; mean(Mean\_Pxx\_ch1(beta\_psd\_i))];

CH1\_psd\_gamma\_lo=[CH1\_psd\_gamma\_lo; mean(Mean\_Pxx\_ch1(gamma\_lo\_psd\_i))];

CH1\_psd\_gamma\_hi=[CH1\_psd\_gamma\_hi; mean(Mean\_Pxx\_ch1(gamma\_hi\_psd\_i))];

CH2\_psd\_delta=[CH2\_psd\_delta; mean(Mean\_Pxx\_ch2(delta\_psd\_i))];

CH2\_psd\_theta=[CH2\_psd\_theta; mean(Mean\_Pxx\_ch2(theta\_psd\_i))];

CH2\_psd\_alpha=[CH2\_psd\_alpha; mean(Mean\_Pxx\_ch2(alpha\_psd\_i))];

CH2\_psd\_beta=[CH2\_psd\_beta; mean(Mean\_Pxx\_ch2(beta\_psd\_i))];

CH2\_psd\_gamma\_lo=[CH2\_psd\_gamma\_lo; mean(Mean\_Pxx\_ch2(gamma\_lo\_psd\_i))];

CH2\_psd\_gamma\_hi=[CH2\_psd\_gamma\_hi; mean(Mean\_Pxx\_ch2(gamma\_hi\_psd\_i))];

% Power Spectrum

eval(strcat('[S\_ch1,f,Serr\_ch1]=mtspectrumc(detrend(seg\_ch1),params);'))

eval(strcat('[S\_ch2,f,Serr\_ch2]=mtspectrumc(detrend(seg\_ch2),params);'))

delta\_i = find(f >= Delta(1) & f < Delta(2));

theta\_i = find(f >= Theta(1) & f < Theta(2));

alpha\_i = find(f >= Alpha(1) & f < Alpha(2));

beta\_i = find(f >= Beta(1) & f < Beta(2));

gamma\_lo\_i = find(f >= Gamma\_lo(1) & f < Gamma\_lo(2));

gamma\_hi\_i = find(f >= Gamma\_hi(1) & f < Gamma\_hi(2));

CH1\_delta=[CH1\_delta; mean(S\_ch1(delta\_i))];

CH1\_theta=[CH1\_theta; mean(S\_ch1(theta\_i))];

CH1\_alpha=[CH1\_alpha; mean(S\_ch1(alpha\_i))];

CH1\_beta=[CH1\_beta; mean(S\_ch1(beta\_i))];

CH1\_gamma\_lo=[CH1\_gamma\_lo; mean(S\_ch1(gamma\_lo\_i))];

CH1\_gamma\_hi=[CH1\_gamma\_hi; mean(S\_ch1(gamma\_hi\_i))];

CH2\_delta=[CH2\_delta; mean(S\_ch2(delta\_i))];

CH2\_theta=[CH2\_theta; mean(S\_ch2(theta\_i))];

CH2\_alpha=[CH2\_alpha; mean(S\_ch2(alpha\_i))];

CH2\_beta=[CH2\_beta; mean(S\_ch2(beta\_i))];

CH2\_gamma\_lo=[CH2\_gamma\_lo; mean(S\_ch2(gamma\_lo\_i))];

CH2\_gamma\_hi=[CH2\_gamma\_hi; mean(S\_ch2(gamma\_hi\_i))];

Coh\_delta=[Coh\_delta; mean(C(delta\_i))];

Coh\_theta=[Coh\_theta; mean(C(theta\_i))];

Coh\_alpha=[Coh\_alpha; mean(C(alpha\_i))];

Coh\_beta=[Coh\_beta; mean(C(beta\_i))];

Coh\_gamma\_lo=[Coh\_gamma\_lo; mean(C(gamma\_lo\_i))];

Coh\_gamma\_hi=[Coh\_gamma\_hi; mean(C(gamma\_hi\_i))];

coh=[coh; C'];

psd\_ch1=[psd\_ch1;Mean\_Pxx\_ch1(1:Fs/2+1)'];

psd\_ch2=[psd\_ch2;Mean\_Pxx\_ch2(1:Fs/2+1)'];

% spect\_ch1=[spect\_ch1; (S\_ch1/mean(S\_ch1(theta\_i)))'];

spect\_ch1=[spect\_ch1; S\_ch1'];

% spect\_ch2=[spect\_ch2; (S\_ch2/mean(S\_ch2(theta\_i)))'];

spect\_ch2=[spect\_ch2; S\_ch2'];

group=[group; g];

end

% Fazer uma discrimina??o das bandas de frequ?ncia

eval(strcat('data',num2str(g),'=data;'))

% eval(strcat('data',num2str(g),'(:,1)=[];'))

%

eval(strcat('data',num2str(g),'=filter(hd, data',num2str(g),');'))

%

% %Separate data in equally sized segments

eval(strcat('n\_seg=size(data',num2str(g),',1)/seg\_length;'))

eval(strcat('data',num2str(g),'\_ch1=zeros(seg\_length,n\_seg);'))

eval(strcat('data',num2str(g),'\_ch2=zeros(seg\_length,n\_seg);'))

for i=1:n\_seg

eval(strcat('data',num2str(g),'\_ch1(:,i)=data',num2str(g),'((i-1)\*seg\_length+1:i\*seg\_length,1);'))

eval(strcat('data',num2str(g),'\_ch2(:,i)=data',num2str(g),'((i-1)\*seg\_length+1:i\*seg\_length,2);'))

end

%

params=struct('tapers',tapers,'pad',pad,'Fs',Fs,'fpass',fpass,'err',err,'trialave',trialave);

%

% eval(strcat('[C\_',num2str(g),',phi\_',num2str(g),',S',num2str(g),'\_ch1xch2,S',num2str(g),'\_ch1,S',num2str(g),'\_ch2,f,confC\_',num2str(g),',phistd\_',num2str(g),',Cerr\_',num2str(g),']=coherencyc(data',num2str(g),'\_ch1,data',num2str(g),'\_ch2,params);'))

% eval(strcat('[S',num2str(g),'\_ch1,f,Serr',num2str(g),'\_ch1]=mtspectrumc(detrend(data',num2str(g),'\_ch1),params);'))

% eval(strcat('[S',num2str(g),'\_ch2,f,Serr',num2str(g),'\_ch2]=mtspectrumc(detrend(data',num2str(g),'\_ch2),params);'))

end

figure(1)

[Group\_Mean,Group\_SEM]=grpstats(coh, group, {'mean','std'});

hold on

for g=1:n\_group

errorbar(f',Group\_Mean(g,:)',Group\_SEM(g,:)')

end

xlwrite(output\_filename,[{'Freq.'} FileName\_array; num2cell(f') num2cell(coh')], 'Coherence')

%title 'Coherency PFC vs Hipp' (canal 1 PFC)

title 'Coherency vHip vs PFC'

xlabel('Frequency (Hz)')

ylabel('Coherency')

hold off

figure(2)

[Group\_Mean,Group\_SEM]=grpstats(spect\_ch1, group, {'mean','std'});

subplot(1,2,1)

hold on

for g=1:n\_group

errorbar(f',Group\_Mean(g,:)',Group\_SEM(g,:)')

end

xlwrite(output\_filename,[{'Freq.'} FileName\_array; num2cell(f') num2cell(spect\_ch1')], 'Power vHip')

%xlwrite(output\_filename,[{'Freq.'} FileName\_array; num2cell(f') num2cell(spect\_ch1')], 'Power PFC')

%title 'Power Frequency Spectrum PFC'

title 'Power Frequency Spectrum vHip'

xlabel('Frequency (Hz)')

ylabel('10\*log10(PowerSpectrum)')

hold off

[Group\_Mean,Group\_SEM]=grpstats(psd\_ch1, group, {'mean','std'});

subplot(1,2,2)

hold on

f\_psd = Fs/2\*linspace(0,1,Fs/2+1);

for g=1:n\_group

errorbar(f\_psd,Group\_Mean(g,:)',Group\_SEM(g,:)')

end

xlwrite(output\_filename,[{'Freq.'} FileName\_array; num2cell((0:1:Fs/2)') num2cell(psd\_ch1')], 'PSD vHip')

%xlwrite(output\_filename,[{'Freq.'} FileName\_array; num2cell((0:1:Fs/2)') num2cell(psd\_ch1')], 'PSD PFC')

%title 'Power Spectral Density PFC'

title 'Power Spectral Density vHip'

xlabel('Frequency (Hz)')

ylabel('10\*log10(PowerSpectralDensity)')

hold off

figure(3)

[Group\_Mean,Group\_SEM]=grpstats(spect\_ch2, group, {'mean','std'});

subplot(1,2,1)

hold on

for g=1:n\_group

errorbar(f',Group\_Mean(g,:)',Group\_SEM(g,:)')

end

xlwrite(output\_filename,[{'Freq.'} FileName\_array; num2cell(f') num2cell(spect\_ch2')], 'Power PFC')

%xlwrite(output\_filename,[{'Freq.'} FileName\_array; num2cell(f') num2cell(spect\_ch2')], 'Power Hipp')

%title 'Power Frequency Spectrum Hipp'

title 'Power Frequency Spectrum PFC'

xlabel('Frequency (Hz)')

ylabel('10\*log10(PowerSpectrum)')

hold off

[Group\_Mean,Group\_SEM]=grpstats(psd\_ch2, group, {'mean','std'});

subplot(1,2,2)

hold on

f\_psd = Fs/2\*linspace(0,1,Fs/2+1);

for g=1:n\_group

errorbar(f\_psd,Group\_Mean(g,:)',Group\_SEM(g,:)')

end

xlwrite(output\_filename,[{'Freq.'} FileName\_array; num2cell(f\_psd') num2cell(psd\_ch2')], 'PSD PFC')

%xlwrite(output\_filename,[{'Freq.'} FileName\_array; num2cell((0:1:125)') num2cell(psd\_ch2')], 'PSD Hipp')

%xlwrite(output\_filename,[{'Freq.'} FileName\_array; num2cell(f\_psd') num2cell(psd\_ch2')], 'PSD Hipp')

%title 'Power Spectral Density Hipp'

title 'Power Spectral Density PFC'

xlabel('Frequency (Hz)')

ylabel('10\*log10(PowerSpectralDensity)')

hold off

figure (4)

hold on

broadband = find(f >= 4 & f <= 40);

%image(f,1:sum(n),coh,'CDataMapping','scaled')

image(f(broadband),1:sum(n),coh(:,broadband),'CDataMapping','scaled')

%title 'Coherency PFC vs Hipp'

title 'Coherency vHip vs PFC'

xlabel('Frequency (Hz)')

ylabel('Frame')

%ylabel('Animals')

hold off

% figure (5)

% subplot(2,2,1) % NAO APARECE???!

% hold on

% image(f(broadband),1:sum(n),spect\_ch1,'CDataMapping','scaled')

% %title 'Power PFC'

% title 'Power vHip'

% xlabel('Frequency (Hz)')

% ylabel('Animals')

% hold off

% subplot(2,2,2) % NAO APARECE???!

% hold on

% image(f(broadband),1:sum(n),spect\_ch2,'CDataMapping','scaled')

% %title 'Power Hipp'

% title 'Power vHip'

% xlabel('Frequency (Hz)')

% ylabel('Animals')

% hold off

figure (5)

subplot(2,2,1) % NAO APARECE???!

hold on

image(f(broadband),1:sum(n),spect\_ch1,'CDataMapping','scaled')

%title 'Power PFC'

title 'Power vHip'

xlabel('Frequency (Hz)')

ylabel('Animals')

hold off

subplot(2,2,2) % NAO APARECE???!

hold on

image(f(broadband),1:sum(n),spect\_ch2,'CDataMapping','scaled')

%title 'Power Hipp'

title 'Power vHip'

xlabel('Frequency (Hz)')

ylabel('Animals')

hold off

subplot(1,2,1)

hold on

broadband = find(f\_psd >= 4 & f\_psd <= 40);

%image(f\_psd,1:sum(n),psd\_ch1,'CDataMapping','scaled')

image(f\_psd(broadband),1:sum(n),psd\_ch1(:,broadband),'CDataMapping','scaled')

%title 'PSD PFC'

title 'PSD vHip'

xlabel('Frequency (Hz)')

ylabel('Frame')

%ylabel('Animals')

hold off

subplot(1,2,2)

hold on

%image(f\_psd,1:sum(n),psd\_ch2,'CDataMapping','scaled')

image(f\_psd(broadband),1:sum(n),psd\_ch2(:,broadband),'CDataMapping','scaled')

%title 'PSD Hipp'

title 'PSD PFC'

xlabel('Frequency (Hz)')

ylabel('Frame')

%ylabel('Animals')

hold off

figure(6)

for c=1:2

eval(strcat('[Group\_Mean\_delta,Group\_SEM\_delta]=grpstats(CH',num2str(c),'\_delta, group, {''mean'',''std''});'))

eval(strcat('[Group\_Mean\_theta,Group\_SEM\_theta]=grpstats(CH',num2str(c),'\_theta, group, {''mean'',''std''});'))

eval(strcat('[Group\_Mean\_alpha,Group\_SEM\_alpha]=grpstats(CH',num2str(c),'\_alpha, group, {''mean'',''std''});'))

eval(strcat('[Group\_Mean\_beta,Group\_SEM\_beta]=grpstats(CH',num2str(c),'\_beta, group, {''mean'',''std''});'))

eval(strcat('[Group\_Mean\_gamma\_lo,Group\_SEM\_gamma\_lo]=grpstats(CH',num2str(c),'\_gamma\_lo, group, {''mean'',''std''});'))

eval(strcat('[Group\_Mean\_gamma\_hi,Group\_SEM\_gamma\_hi]=grpstats(CH',num2str(c),'\_gamma\_hi, group, {''mean'',''std''});'))

subplot(2,5,(c-1)\*5+1) %Delta

hold on

for g=1:n\_group

errorbar(g,Group\_Mean\_delta(g),Group\_SEM\_delta(g),'x')

end

title 'Delta (0.5-3 Hz)'

ylabel('Power (uV^2)')

xlabel('Group')

hold off

subplot(2,5,(c-1)\*5+2) %Theta

hold on

for g=1:n\_group

errorbar(g,Group\_Mean\_theta(g),Group\_SEM\_theta(g),'x')

end

title 'Theta (4-12 Hz)'

ylabel('Power (uV^2)')

xlabel('Group')

hold off

% subplot(2,6,(c-1)\*6+3) %Alpha

% hold on

% for g=1:n\_group

% errorbar(g,Group\_Mean\_alpha(g),Group\_SEM\_alpha(g),'x')

% end

% title 'Alpha (8-12 Hz)'

% ylabel('Power (uV^2)')

% xlabel('Group')

% hold off

subplot(2,5,(c-1)\*5+3) %Beta

hold on

for g=1:n\_group

errorbar(g,Group\_Mean\_beta(g),Group\_SEM\_beta(g),'x')

end

title 'Beta (13-19 Hz)'

ylabel('Power (uV^2)')

xlabel('Group')

hold off

subplot(2,5,(c-1)\*5+4) %Gamma\_low

hold on

for g=1:n\_group

errorbar(g,Group\_Mean\_gamma\_lo(g),Group\_SEM\_gamma\_lo(g),'x')

end

title 'Low Gamma (20-39 Hz)'

ylabel('Power (uV^2)')

xlabel('Group')

hold off

subplot(2,5,(c-1)\*5+5) %Gamma\_high

hold on

for g=1:n\_group

errorbar(g,Group\_Mean\_gamma\_hi(g),Group\_SEM\_gamma\_hi(g),'x')

end

title 'High Gamma (40-100 Hz)'

ylabel('Power (uV^2)')

xlabel('Group')

hold off

end

xlwrite(output\_filename, [{'Animal'} {'Delta'} {'Theta'} {'Alpha'} {'Beta'} {'LoGamma'} {'HiGamma'}; ...

FileName\_array' num2cell([CH1\_delta CH1\_theta CH1\_alpha CH1\_beta CH1\_gamma\_lo CH1\_gamma\_hi])], 'Power Bands vHip')

xlwrite(output\_filename,[{'Animal'} {'Delta'} {'Theta'} {'Alpha'} {'Beta'} {'LoGamma'} {'HiGamma'}; ...

FileName\_array' num2cell([CH2\_delta CH2\_theta CH2\_alpha CH2\_beta CH2\_gamma\_lo CH2\_gamma\_hi])], 'Power Bands PFC')

%xlwrite(output\_filename, [{'Animal'} {'Delta'} {'Theta'} {'Beta'} {'LoGamma'} {'HiGamma'}; ...

% FileName\_array' num2cell([CH1\_delta CH1\_theta CH1\_beta CH1\_gamma\_lo CH1\_gamma\_hi])], 'Power Bands vHip')

%xlwrite(output\_filename,[{'Animal'} {'Delta'} {'Theta'} {'Beta'} {'LoGamma'} {'HiGamma'}; ...

% FileName\_array' num2cell([CH2\_delta CH2\_theta CH2\_beta CH2\_gamma\_lo CH2\_gamma\_hi])], 'Power Bands PFC')

figure(7)

for c=1:2

eval(strcat('[Group\_Mean\_delta,Group\_SEM\_delta]=grpstats(CH',num2str(c),'\_psd\_delta, group, {''mean'',''std''});'))

eval(strcat('[Group\_Mean\_theta,Group\_SEM\_theta]=grpstats(CH',num2str(c),'\_psd\_theta, group, {''mean'',''std''});'))

eval(strcat('[Group\_Mean\_alpha,Group\_SEM\_alpha]=grpstats(CH',num2str(c),'\_psd\_alpha, group, {''mean'',''std''});'))

eval(strcat('[Group\_Mean\_beta,Group\_SEM\_beta]=grpstats(CH',num2str(c),'\_psd\_beta, group, {''mean'',''std''});'))

eval(strcat('[Group\_Mean\_gamma\_lo,Group\_SEM\_gamma\_lo]=grpstats(CH',num2str(c),'\_psd\_gamma\_lo, group, {''mean'',''std''});'))

eval(strcat('[Group\_Mean\_gamma\_hi,Group\_SEM\_gamma\_hi]=grpstats(CH',num2str(c),'\_psd\_gamma\_hi, group, {''mean'',''std''});'))

subplot(2,5,(c-1)\*5+1) %Delta

hold on

for g=1:n\_group

errorbar(g,Group\_Mean\_delta(g),Group\_SEM\_delta(g),'x')

end

title 'Delta (0.5-3 Hz)'

ylabel('PSD (uV^2/Hz)')

xlabel('Group')

hold off

subplot(2,5,(c-1)\*5+2) %Theta

hold on

for g=1:n\_group

errorbar(g,Group\_Mean\_theta(g),Group\_SEM\_theta(g),'x')

end

title 'Theta (4-12 Hz)'

ylabel('PSD (uV^2/Hz)')

xlabel('Group')

hold off

% subplot(2,6,(c-1)\*6+3) %Alpha

% hold on

% for g=1:n\_group

% errorbar(g,Group\_Mean\_alpha(g),Group\_SEM\_alpha(g),'x')

% end

% title 'Alpha (8-12 Hz)'

% ylabel('PSD (uV^2/Hz)')

% xlabel('Group')

% hold off

subplot(2,5,(c-1)\*5+3) %Beta

hold on

for g=1:n\_group

errorbar(g,Group\_Mean\_beta(g),Group\_SEM\_beta(g),'x')

end

title 'Beta (13-19 Hz)'

ylabel('PSD (uV^2/Hz)')

xlabel('Group')

hold off

subplot(2,5,(c-1)\*5+4) %Gamma\_low

hold on

for g=1:n\_group

errorbar(g,Group\_Mean\_gamma\_lo(g),Group\_SEM\_gamma\_lo(g),'x')

end

title 'Low Gamma (20-39 Hz)'

ylabel('PSD (uV^2/Hz)')

xlabel('Group')

hold off

subplot(2,5,(c-1)\*5+5) %Gamma\_high

hold on

for g=1:n\_group

errorbar(g,Group\_Mean\_gamma\_hi(g),Group\_SEM\_gamma\_hi(g),'x')

end

title 'High Gamma (40-100 Hz)'

ylabel('PSD (uV^2/Hz)')

xlabel('Group')

hold off

end

xlwrite(output\_filename, [{'Animal'} {'Delta'} {'Theta'} {'Alpha'} {'Beta'} {'LoGamma'} {'HiGamma'}; ...

FileName\_array' num2cell([CH1\_psd\_delta CH1\_psd\_theta CH1\_psd\_alpha CH1\_psd\_beta CH1\_psd\_gamma\_lo CH1\_psd\_gamma\_hi])], 'PSD Bands vHip')

xlwrite(output\_filename,[{'Animal'} {'Delta'} {'Theta'} {'Alpha'} {'Beta'} {'LoGamma'} {'HiGamma'}; ...

FileName\_array' num2cell([CH2\_psd\_delta CH2\_psd\_theta CH2\_psd\_alpha CH2\_psd\_beta CH2\_psd\_gamma\_lo CH2\_psd\_gamma\_hi])], 'PSD Bands PFC')

%xlwrite(output\_filename, [{'Animal'} {'Delta'} {'Theta'} {'Beta'} {'LoGamma'} {'HiGamma'}; ...

% FileName\_array' num2cell([CH1\_psd\_delta CH1\_psd\_theta CH1\_psd\_beta CH1\_psd\_gamma\_lo CH1\_psd\_gamma\_hi])], 'PSD Bands PFC')

%xlwrite(output\_filename,[{'Animal'} {'Delta'} {'Theta'} {'Beta'} {'LoGamma'} {'HiGamma'}; ...

% FileName\_array' num2cell([CH2\_psd\_delta CH2\_psd\_theta CH2\_psd\_beta CH2\_psd\_gamma\_lo CH2\_psd\_gamma\_hi])], 'PSD Bands Hipp')

figure(8)

eval(strcat('[Group\_Mean\_delta,Group\_SEM\_delta]=grpstats(Coh\_delta, group, {''mean'',''std''});'))

eval(strcat('[Group\_Mean\_theta,Group\_SEM\_theta]=grpstats(Coh\_theta, group, {''mean'',''std''});'))

eval(strcat('[Group\_Mean\_alpha,Group\_SEM\_alpha]=grpstats(Coh\_alpha, group, {''mean'',''std''});'))

eval(strcat('[Group\_Mean\_beta,Group\_SEM\_beta]=grpstats(Coh\_beta, group, {''mean'',''std''});'))

eval(strcat('[Group\_Mean\_gamma\_lo,Group\_SEM\_gamma\_lo]=grpstats(Coh\_gamma\_lo, group, {''mean'',''std''});'))

eval(strcat('[Group\_Mean\_gamma\_hi,Group\_SEM\_gamma\_hi]=grpstats(Coh\_gamma\_hi, group, {''mean'',''std''});'))

xlwrite(output\_filename,[{'Animal'} {'Delta'} {'Theta'} {'Beta'} {'LoGamma'} {'HiGamma'}; ...

FileName\_array' num2cell([Coh\_delta Coh\_theta Coh\_beta Coh\_gamma\_lo Coh\_gamma\_hi])], 'Coherence Bands')

subplot(1,5,1) %Delta

hold on

for g=1:n\_group

errorbar(g,Group\_Mean\_delta(g),Group\_SEM\_delta(g),'x')

end

title 'Delta (0.5-3 Hz)'

ylabel('Coherence (0-1)')

xlabel('Group')

hold off

subplot(1,5,2) %Theta

hold on

for g=1:n\_group

errorbar(g,Group\_Mean\_theta(g),Group\_SEM\_theta(g),'x')

end

title 'Theta (4-12 Hz)'

ylabel('Coherence (0-1)')

xlabel('Group')

hold off

% subplot(1,6,3) %Alpha

% hold on

% for g=1:n\_group

% errorbar(g,Group\_Mean\_alpha(g),Group\_SEM\_alpha(g),'x')

% end

% title 'Alpha (8-12 Hz)'

% ylabel('Coherence (0-1)')

% xlabel('Group')

% hold off

subplot(1,5,3) %Beta

hold on

for g=1:n\_group

errorbar(g,Group\_Mean\_beta(g),Group\_SEM\_beta(g),'x')

end

title 'Beta (13-19 Hz)'

ylabel('Coherence (0-1)')

xlabel('Group')

hold off

subplot(1,5,4) %Gamma Low

hold on

for g=1:n\_group

errorbar(g,Group\_Mean\_gamma\_lo(g),Group\_SEM\_gamma\_lo(g),'x')

end

title 'Low Gamma (20-39 Hz)'

ylabel('Coherence (0-1)')

xlabel('Group')

hold off

subplot(1,5,5) %Gamma High

hold on

for g=1:n\_group

errorbar(g,Group\_Mean\_gamma\_hi(g),Group\_SEM\_gamma\_hi(g),'x')

end

title 'High Gamma (40-100 Hz)'

ylabel('Coherence (0-1)')

xlabel('Group')

hold off

%ULTIMOS GRAFICOS DO POWER QUE NAO ESTAVAM A SER REPRESENTADOS

% figure (9)

% subplot(1,2,1) % NAO APARECE???!

% hold on

% image(f(broadband),1:sum(n),spect\_ch1,'CDataMapping','scaled')

% %title 'Power PFC'

% title 'Power vHip'

% xlabel('Frequency (Hz)')

% ylabel('Animals')

% hold off

% subplot(1,2,2) % NAO APARECE???!

% hold on

% image(f(broadband),1:sum(n),spect\_ch2,'CDataMapping','scaled')

% %title 'Power Hipp'

% title 'Power vHip'

% xlabel('Frequency (Hz)')

% ylabel('Animals')

% hold off

% save 'all\_animals.mat' group coh spect\_ch1 spect\_ch2

for i=1:n2 [FileName,PathName,FilterIndex]=uigetfile({'.txt'}, 'Group 2'); file=strcat(PathName,FileName); [data]=importdata(file); % data(:,2) = mat2gray(data(:,2)) - mean(mat2gray(data(:,2))); % data(:,3) = mat2gray(data(:,3)) - mean(mat2gray(data(:,3))); % data=normc(data); data2=[data2;data]; end uigetfile pause time=data1(:,1); data1(:,1)=[]; uiimport pause data2(:,1)=[];

% Line Filter

% hd=LineFilter;

% data1=filter(hd, data1);

% data2=filter(hd, data2);

% data1(:,1) = mat2gray(data1(:,1)) - mean(mat2gray(data1(:,1)));

% data1(:,2) = mat2gray(data1(:,2)) - mean(mat2gray(data1(:,2)));

% data2(:,1) = mat2gray(data2(:,1)) - mean(mat2gray(data2(:,1)));

% data2(:,2) = mat2gray(data2(:,2)) - mean(mat2gray(data2(:,2)));

% tapers=[3 5];

% pad=0;

% Fs=250;

% fpass=[0 100];

% err= [2 0.05];

% trialave=1;

%Separate data in equally sized segments

% seg\_length=2\*Fs;

% n\_seg=size(data1,1)/seg\_length;

% data1\_ch1=zeros(seg\_length,n\_seg);

% data1\_ch2=zeros(seg\_length,n\_seg);

% for i=1:n\_seg

% data1\_ch1(:,i)=data1((i-1)\*seg\_length+1:i\*seg\_length,1);

% data1\_ch2(:,i)=data1((i-1)\*seg\_length+1:i\*seg\_length,2);

% end

%

% n\_seg=size(data2,1)/seg\_length;

% data2\_ch1=zeros(seg\_length,n\_seg);

% data2\_ch2=zeros(seg\_length,n\_seg);

% for i=1:n\_seg

% data2\_ch1(:,i)=data2((i-1)\*seg\_length+1:i\*seg\_length,1);

% data2\_ch2(:,i)=data2((i-1)\*seg\_length+1:i\*seg\_length,2);

% end

% params=struct('tapers',tapers,'pad',pad,'Fs',Fs,'fpass',fpass,'err',err,'trialave',trialave);

%

% [C\_init,phi\_init,Sinit\_ch1xch2,Sinit\_ch1,Sinit\_ch2,f,confC\_init,phistd\_init,Cerr\_init]=coherencyc(data1\_ch1,data1\_ch2,params);

% [C\_end,phi\_end,Send\_ch1xch2,Send\_ch1,Send\_ch2,f,confC\_end,phistd\_end,Cerr\_end]=coherencyc(data2\_ch1,data2\_ch2,params);

% figure(2)

% % eval(strcat('fill([f f(end:-1:1)],[Cerr\_1(1,:) Cerr\_1(2,end:-1:1)],[0.5 0.5 0.1])'))

% hold on

% for g=1:n\_group

% eval(strcat('fill([f f(end:-1:1)],[Cerr\_',num2str(g),'(1,:) Cerr\_',num2str(g),'(2,end:-1:1)],[0.5 0.5 0.5])'))

% end

% for g=1:n\_group

% eval(strcat('plot(f,C\_',num2str(g),',''-b'',''LineWidth'',1)'))

% end

% % plot(f,C\_end,'-k','LineWidth',1)

% title 'Coherency PFC vs Hipp'

% xlabel('Frequency (Hz)')

% ylabel('Coherency')

% hold off

%

% % [S1\_ch1,f,Serr1\_ch1]=mtspectrumc(detrend(data1\_ch1),params);

% % [S1\_ch2,f,Serr1\_ch2]=mtspectrumc(detrend(data1\_ch2),params);

% % [S2\_ch1,f,Serr2\_ch1]=mtspectrumc(detrend(data2\_ch1),params);

% % [S2\_ch2,f,Serr2\_ch2]=mtspectrumc(detrend(data2\_ch2),params);

%

% figure(3)

% % fill([f f(end:-1:1)],[10\*log10(Serr1\_ch1(1,:)) 10\*log10(Serr1\_ch1(2,end:-1:1))],[1 0.5 0.5])

% hold on

% for g=1:n\_group

% eval(strcat('fill([f f(end:-1:1)],[10\*log10(Serr',num2str(g),'\_ch1(1,:)) 10\*log10(Serr',num2str(g),'\_ch1(2,end:-1:1))],[0.5 0.5 0.5])'))

% eval(strcat('plot(f,10\*log10(S',num2str(g),'\_ch1),''-r'',''LineWidth'',1)'))

% end

% % plot(f,10\*log10(S2\_ch1),'-k','LineWidth',1)

% title 'PFC init vs end'

% xlabel('Frequency (Hz)')

% ylabel('10\*log10(PowerSpectrum)')

% hold off

%

% figure(4)

% % fill([f f(end:-1:1)],[10\*log10(Serr1\_ch2(1,:)) 10\*log10(Serr1\_ch2(2,end:-1:1))],[1 0.5 0.2])

% hold on

% for g=1:n\_group

% eval(strcat('fill([f f(end:-1:1)],[10\*log10(Serr',num2str(g),'\_ch2(1,:)) 10\*log10(Serr',num2str(g),'\_ch2(2,end:-1:1))],[0.5 0.5 0.5])'))

% eval(strcat('plot(f,10\*log10(S',num2str(g),'\_ch2),''-w'',''LineWidth'',1)'))

% end

% % plot(f,10\*log10(S2\_ch2),'-k','LineWidth',1)

% title 'Hipp init vs end'

% xlabel('Frequency (Hz)')

% ylabel('10\*log10(PowerSpectrum)')

% hold off