Simulate Partial Directed Coherence – Schizophrenia EEG

Alireza KHODABAKHSH

24/NOV/2021

PDC estimation – Main Function

- [ePDC, f, p_e]=CONNECTIVITY_PDC(
- folder_address, file_name, fc, nfft, idMode, p, show
-)
- fc=128
- nfft=7680
- idMode=0 : Least Squares covariance
- p=5
- show
- ePDC
- p_e

PDC estimation – Read Data

```
%% Read DATA %%
    Y=schizo_eeg_converter_func(folder_address, file_name);

function EEG=schizo_eeg_converter_func(address, name)
    X=importdata([address,'\' ,name]);
    num_sample=numel(X)/16;
    EEG=zeros(16,num_sample);
    for j=1:16
        Start=(j-1)*num_sample + 1;
        End=j*num_sample;
        EEG(j,:)=X(Start:End,1)';
    end
end
```

PDC estimation – Order Estimator

- [ePDC, f, p_e]=CONNECTIVITY_PDC(
- folder_address, file_name, fc, nfft, idMode, p, show
-)

```
%% Estimate Order %%
    pcrit='aic';
    if pcrit(1)=='a' || pcrit(1)=='m'
        [pottaic,pottmdl,aic,mdl] = mos_idMVAR(Y,20,idMode);
        if pcrit(1)=='a', p_e=pottaic; else p_e=pottmdl; end
    else
        p_e=pcrit; % p estimation
    end
```

PDC estimation – Estimate Coeff MVAR

- [ePDC, f, p_e]=CONNECTIVITY_PDC(
- folder_address, file_name, fc, nfft, idMode, p, show
-)
- %% Estimate Coeeff %%
- [eAm,eSu,~,~]=idMVAR(Y,p,idMode);

- [Am,S,Yp,Up]=idMVAR(Y,p,Mode) ➔ **Use INTERNET**
- % Y, M*N matrix of time series (each time series is in a row)
- % p, model order
- % Mode, determines estimation algorithm (0:builtin least squares, else other methods [see mvar.m from biosig package])

PDC estimation – Estimate PDC from MVAR

- [ePDC, f, p_e]=CONNECTIVITY_PDC(
- folder_address, file_name, fc, nfft, idMode, p, show
-)
- %% %%% Estimated spectral functions
- [~,~,~,gpdc2,~,~,~,~,~,~,f] = fdMVAR(eAm,eSu,nfft,fc);
- ePDC=abs(gpdc2).^2; % partial directed coherence
- [DC,DTF,PDC,GPDC,COH,PCOH,PCOH2,H,S,P,f] = fdMVAR(Am,Su,N,Fs)→ **INTERNET**
- % Am=[A(1)...A(p)]: M*pM matrix of the MVAR model coefficients (strictly causal model)
- % Su: M*M covariance matrix of the input noises
- % N= number of points for calculation of the spectral functions (nfft)
- % Fs= sampling frequency
- % GPDC= Generalized Partial Directed Coherence
- PDC= Partial Directed Coherence

PDC estimation – Plot PDC (Frequency Domain)

```
[ePDC, f, p_e]=CONNECTIVITY_PDC(
folder_address, file_name, fc, nfft, idMode, p, show
)
%% SHOW %%
 %% reshape in order to plot %%
        mydata=zeros(4,4,nfft,16);
        k=1;
        for i=0:3
            for j=0:3
                mydata(:,:,:,k)=ePDC((4*i+1):(4*i+4),...
                                     (4*j+1):(4*j+4),...
                                      :             );
                k=k+1;
            end
        end
```

PDC estimation – Computational Cost

```
[ePDC, f, p_e]=CONNECTIVITY_PDC(
folder_address, file_name, fc, nfft, idMode, p, show
)
%% Plot %%
        M=size(eAm,1);
        for k=1:16
            figure(k)
            sgtitle(['Block', num2str(k) ,'th'])
            q=1;
            for i=1:M/4
                for j=1:M/4
                    subplot(4,4,q);
                    plot(f, squeeze(mydata(i,j,:,k)),'r'); hold on;
                    grid on;
                    axis([0 fc/2 -0.05 1.05]);
                    title(['PDC : ',num2str(j),'\rightarrow',num2str(i)])
                    q=q+1;
                end
            end
        end
```

PDC estimation – Convert PDC signals to Matrix

```
• [ePDC, f, p_e]=CONNECTIVITY_PDC(
• folder_address, file_name, fc, nfft, idMode, p, show
• )
• %% Connectivity Matrix %%
• freq_range=[1,4;4,7;8,13;14,30;30,64];
• connectivity_matrix=zeros(16,16,5);
• for k=1:5
•     for i=1:16
•         for j=1:16
•             freq_indx=(find(f>=freq_range(k,1) & f<=freq_range(k,2)));
•             connectivity_matrix(i,j,k)=mean(squeeze(ePDC(i,j,freq_indx)));
•             connectivity_matrix(:,:,k)=connectivity_matrix(:,:,k)-
  diag(diag(connectivity_matrix(:,:,k)));
•         end
•     end
• end
```

PDC estimation – Save and Transfer to Python

- [ePDC, f, p_e]=CONNECTIVITY_PDC(
- folder_address, file_name, fc, nfft, idMode, p, show
-)
- %% Save and Plot %%
- figure()
- imagesc(connectivity_matrix(:,:,3))
- title('Normal/Subject_1/Alpha')
- save('A:\Term 5th\Paper\MVAR&PDC - MatLab\connectivity matrix for python\subject_schizo_2')

Matrix & Graph – import Libraries

- `!pip install nilearn`
- `import tensorflow as tf`
- `import tensorflow.compat.v1 as tf`
- `tf.disable_v2_behavior()`
- `from tensorflow.keras.datasets import mnist`
- `import matplotlib.pyplot as plt`
- `import numpy as np`
- `import pandas as pd`
- `import glob`
- `import os`
- `import sklearn.preprocessing`
- `from nilearn.input_data import NiftiMapsMasker`
- `from nilearn.connectome import ConnectivityMeasure`
- `from nilearn import plotting`
- `import scipy.io`

Matrix & Graph – Manual Parameters

- #parameters
- type_class='normal' # 'normal'/'schizo'
- sub=2 # 1/2
- band_sel='alpha'

Matrix & Graph – Load Data from Google Drive and Plot Matrix

- `X = scipy.io.loadmat('subject_'+type_class+'_'+str(sub)+'.mat')`
- `band={'delta':0 , 'theta':1 , 'alpha' : 2 , 'beta' : 3, 'gamma' : 4}`
- `labels=['1 - F7','2 - F3','3 - F4','4 - F8','5 - T3','6 - C3','7 - Cz','8 - C4','9 - T4','10 - T5','11 - P3','12 - Pz','13 - P4','14 - T6','15 - O1','16 - O2’]`
- `X= X['connectivity_matrix']`
- `plotting.plot_matrix(X[:,:,band[band_sel]], labels=labels, colorbar=True)`
- `plt.title('Class: {} - Subject: {}th - Band: {}'.format(type_class, sub, band_sel))`

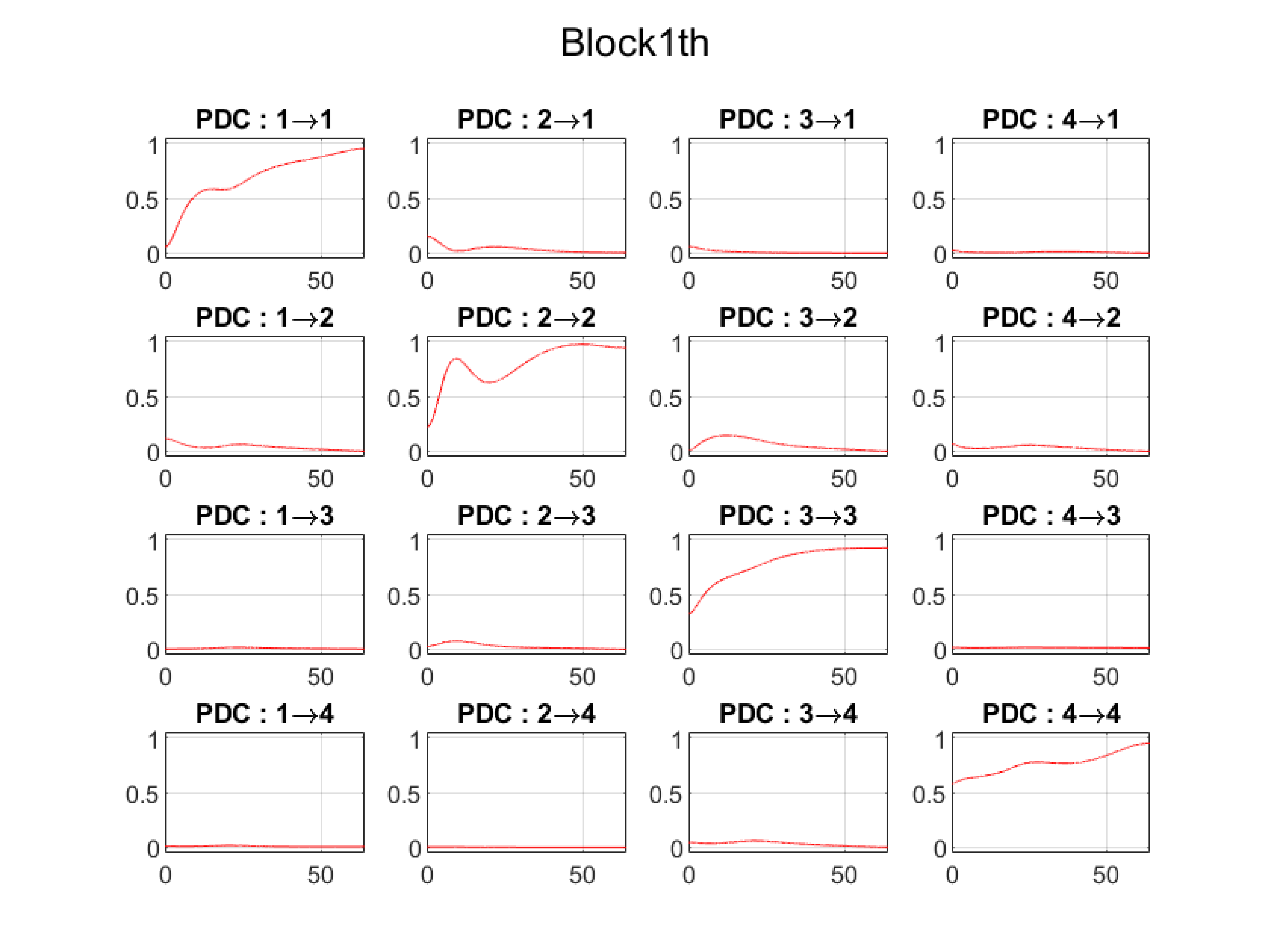
Matrix & Graph – Plot Graph

- coords=[(-68.423 , 49.871 , -7.4895) ,
- (-48.2 , 57.551 , 39.87) ,
- (48.143 , 57.584 , 39.892) ,
- (68.384 , 49.927 , -7.4851),
- (-84.539 , 0 , -8.8451) ,
- (-63.171 , 0 , 56.872),
- (0 , 0 , 85) ,
- (63.167 , 0 , 56.876) ,
- (84.539 , 0 , -8.8451),
- (-49.709 , -68.691, -5.9589),
- (-48.2 , -57.551, 39.87),
- (0 , -60.738, 59.463),
- (48.143 , -57.584, 39.892),
- (49.669 , -68.721, -5.953) ,
- (-26.133, -80.784, -4.0011),
- (26.133 , -80.784, -4.0011)]
- plotting.plot_connectome(X[:,:,band[band_sel]], coords,
- edge_threshold="80%", colorbar=True,
- title='Class: {} - Subject: {}th - Band: {}'.format(type_class, sub, band_sel))
- plotting.show()

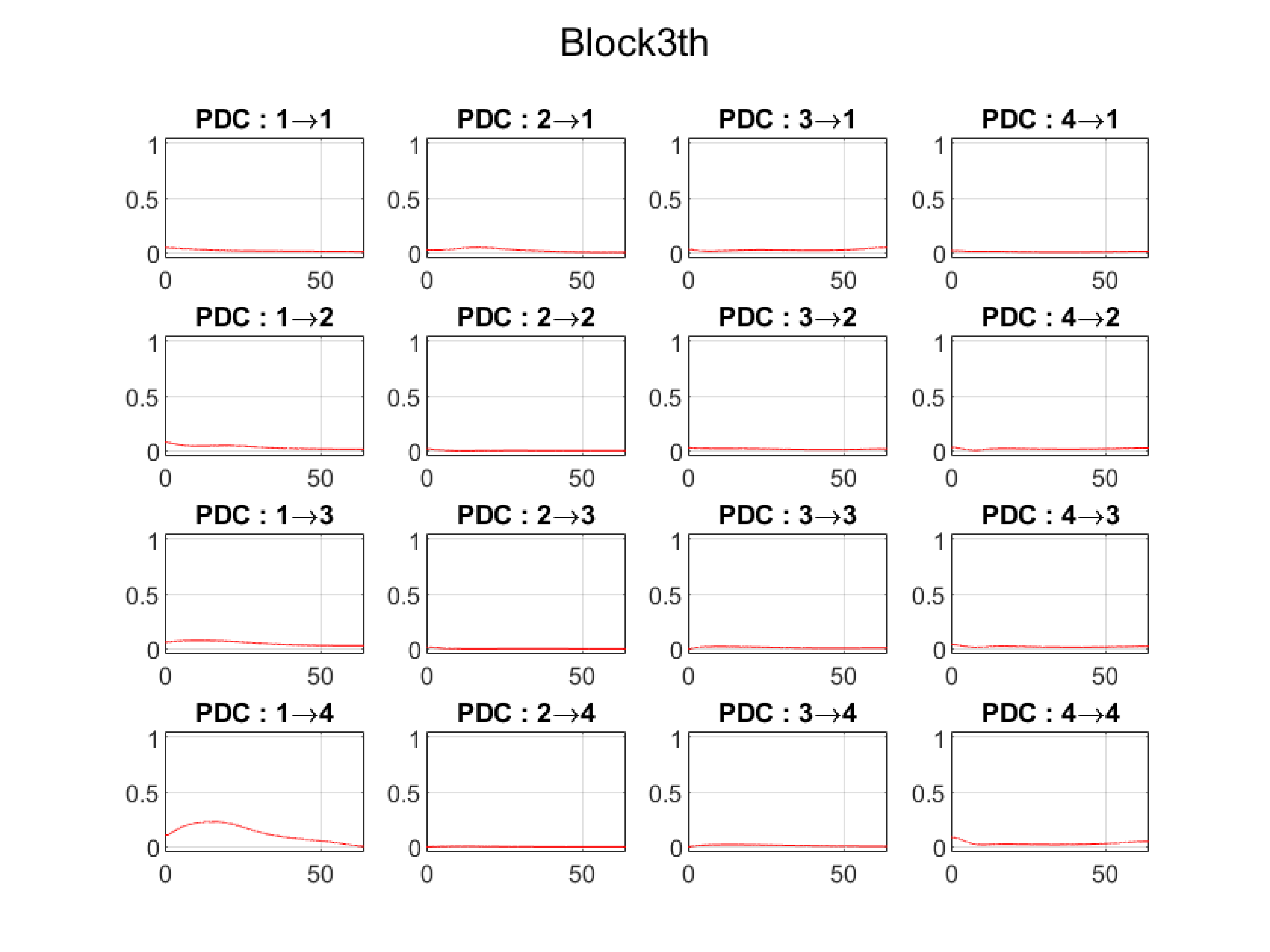
Matrix & Graph – View 3D Graph

- view = plotting.view_connectome(X[:,:,band[band_sel]], coords, edge_threshold='80%')
-
  # In a Jupyter notebook, if ``view`` is the output of a cell, it will
- # be displayed below the cell
- view

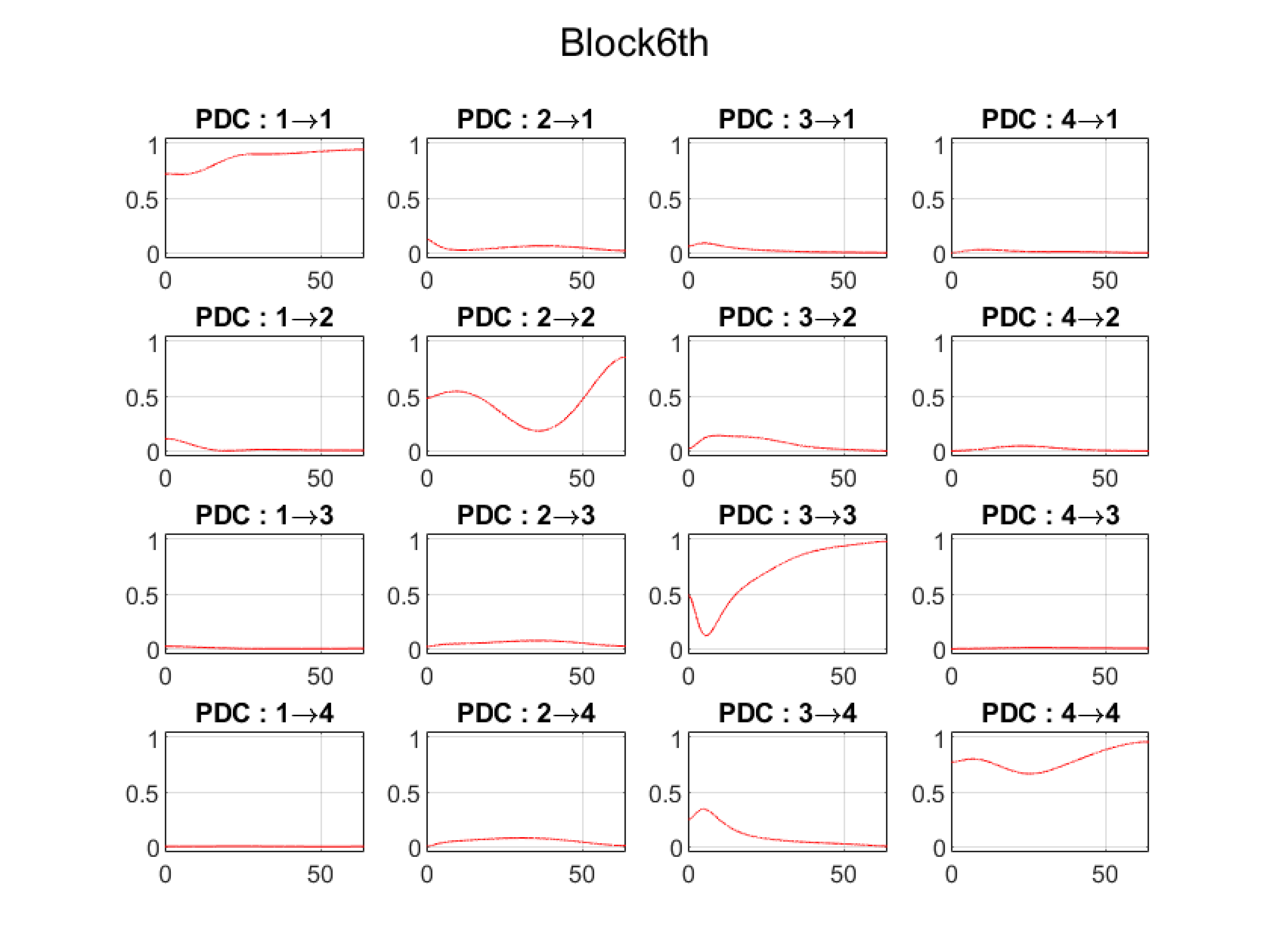
Results


Block1th
PDC : 1→1
PDC : 2→1
PDC : 3→1
PDC : 4→1
PDC : 1→2
PDC : 2→2
PDC : 3→2
PDC : 4→2
PDC : 1→3
PDC : 2→3
PDC : 3→3
PDC : 4→3
PDC : 1→4
PDC : 2→4
PDC : 3→4
PDC : 4→4
0
0.5
1
0
50

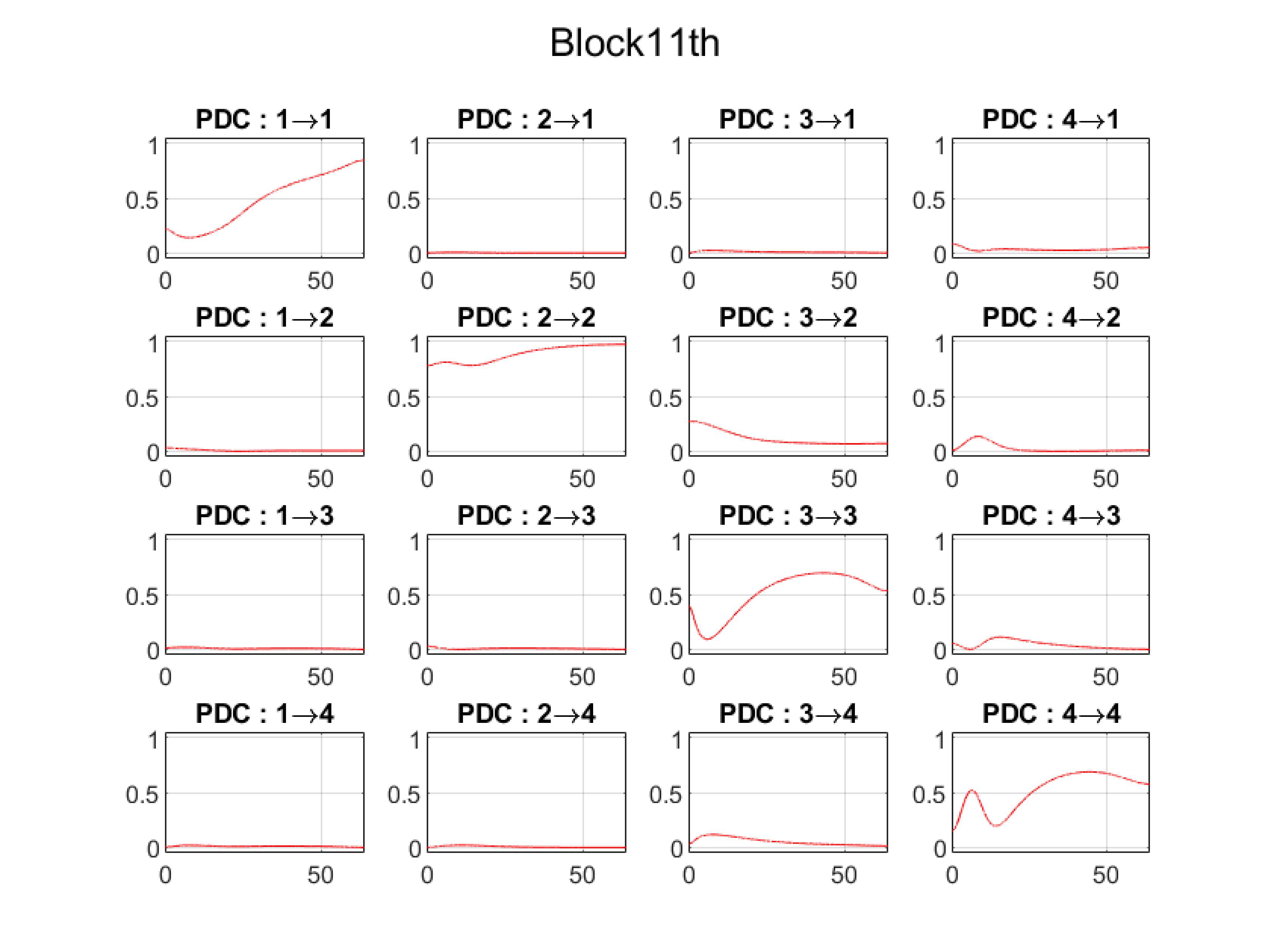
Results


Block3th
PDC : 1→1
PDC : 2→1
PDC : 3→1
PDC : 4→1
PDC : 1→2
PDC : 2→2
PDC : 3→2
PDC : 4→2
PDC : 1→3
PDC : 2→3
PDC : 3→3
PDC : 4→3
PDC : 1→4
PDC : 2→4
PDC : 3→4
PDC : 4→4
1
0.5
0
50

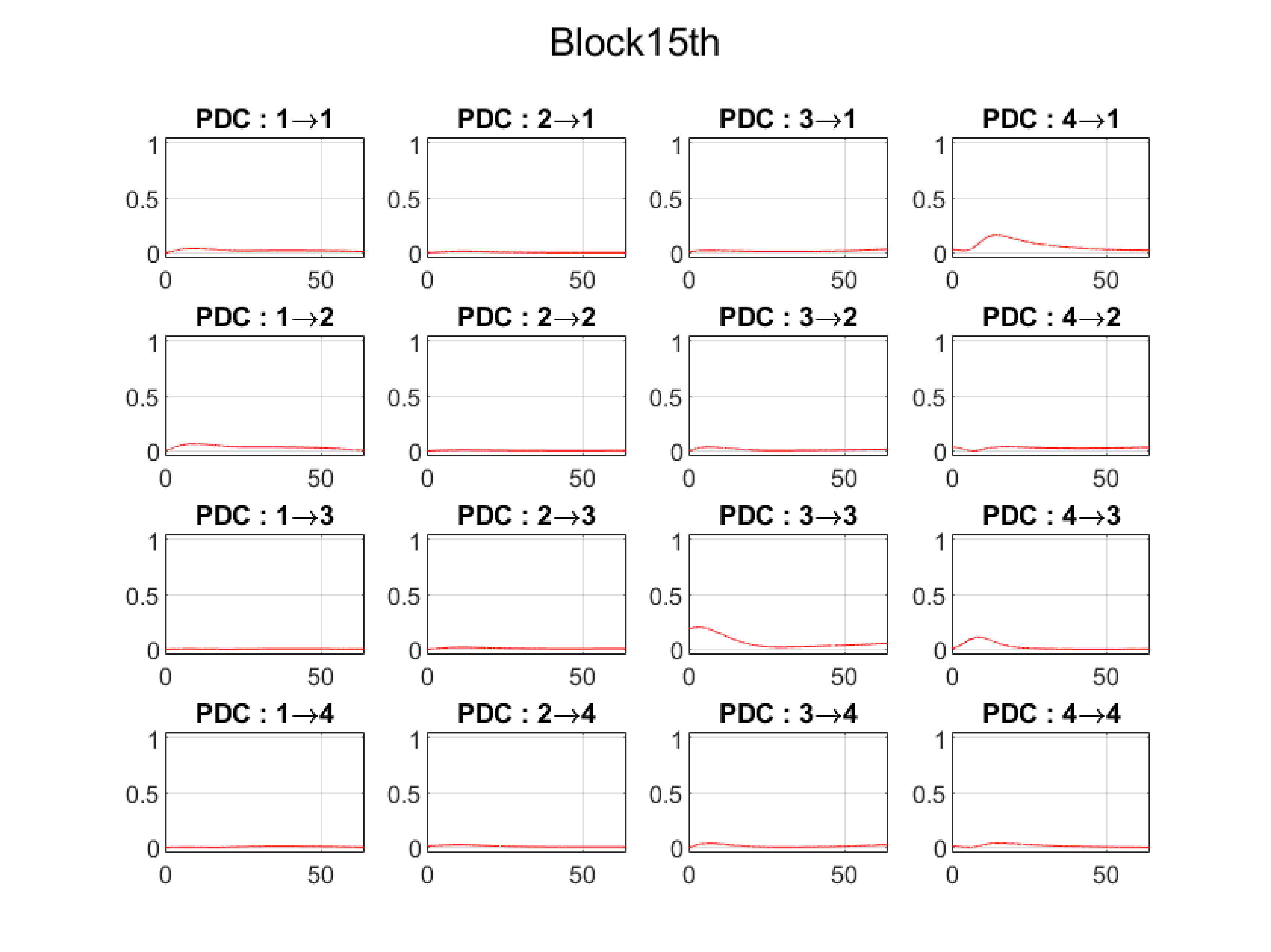
Results


Block6th
PDC : 1→1
PDC : 2→1
PDC : 3→1
PDC : 4→1
PDC : 1→2
PDC : 2→2
PDC : 3→2
PDC : 4→2
PDC : 1→3
PDC : 2→3
PDC : 3→3
PDC : 4→3
PDC : 1→4
PDC : 2→4
PDC : 3→4
PDC : 4→4
0
0.5
1
0
50

Results

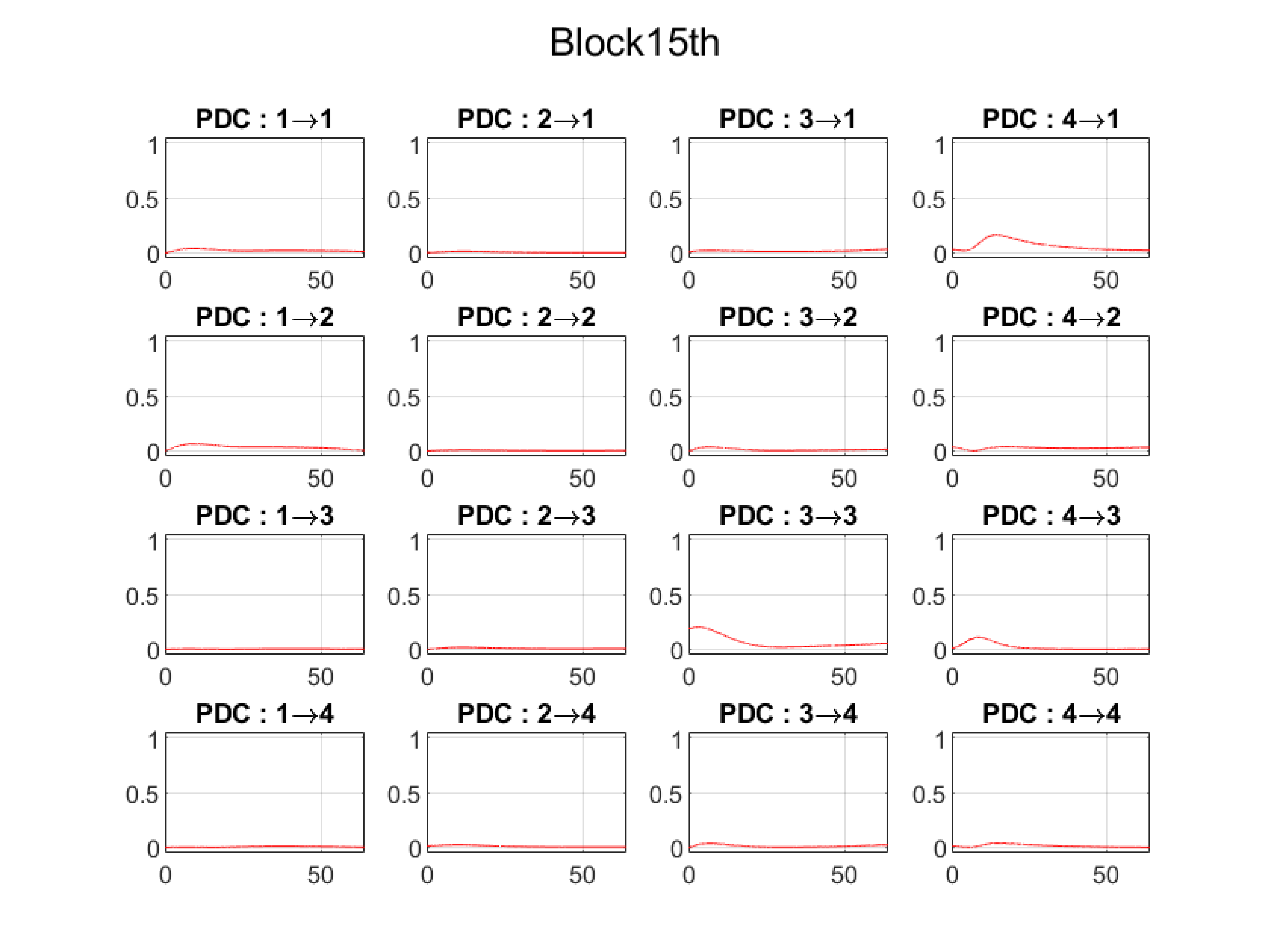

Block11th
PDC : 1→1
PDC : 2→1
PDC : 3→1
PDC : 4→1
PDC : 1→2
PDC : 2→2
PDC : 3→2
PDC : 4→2
PDC : 1→3
PDC : 2→3
PDC : 3→3
PDC : 4→3
PDC : 1→4
PDC : 2→4
PDC : 3→4
PDC : 4→4

Results

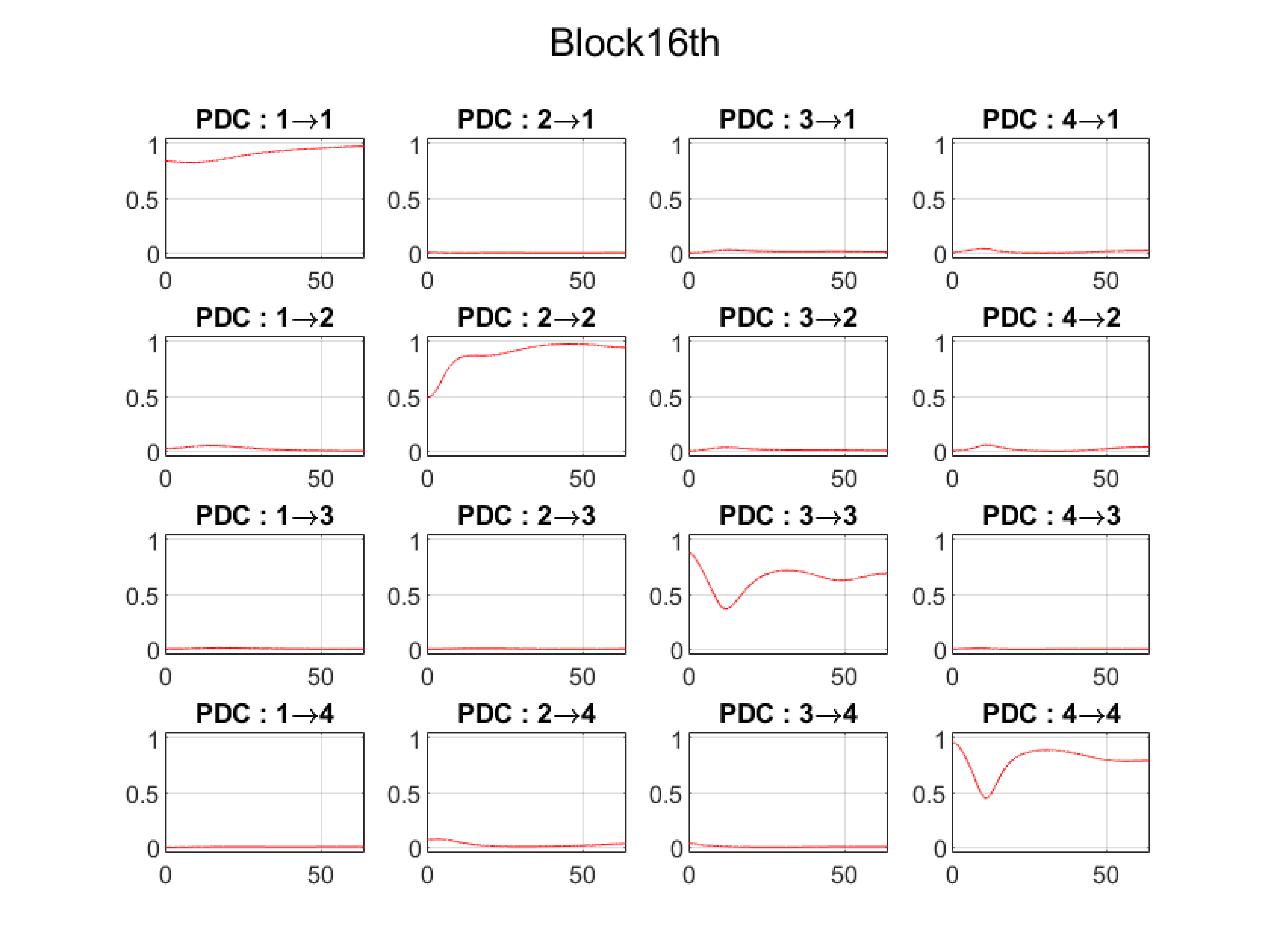


Block15th
PDC : 1→1
PDC : 2→1
PDC : 3→1
PDC : 4→1
PDC : 1→2
PDC : 2→2
PDC : 3→2
PDC : 4→2
PDC : 1→3
PDC : 2→3
PDC : 3→3
PDC : 4→3
PDC : 1→4
PDC : 2→4
PDC : 3→4
PDC : 4→4
1
0.5
0
0
50

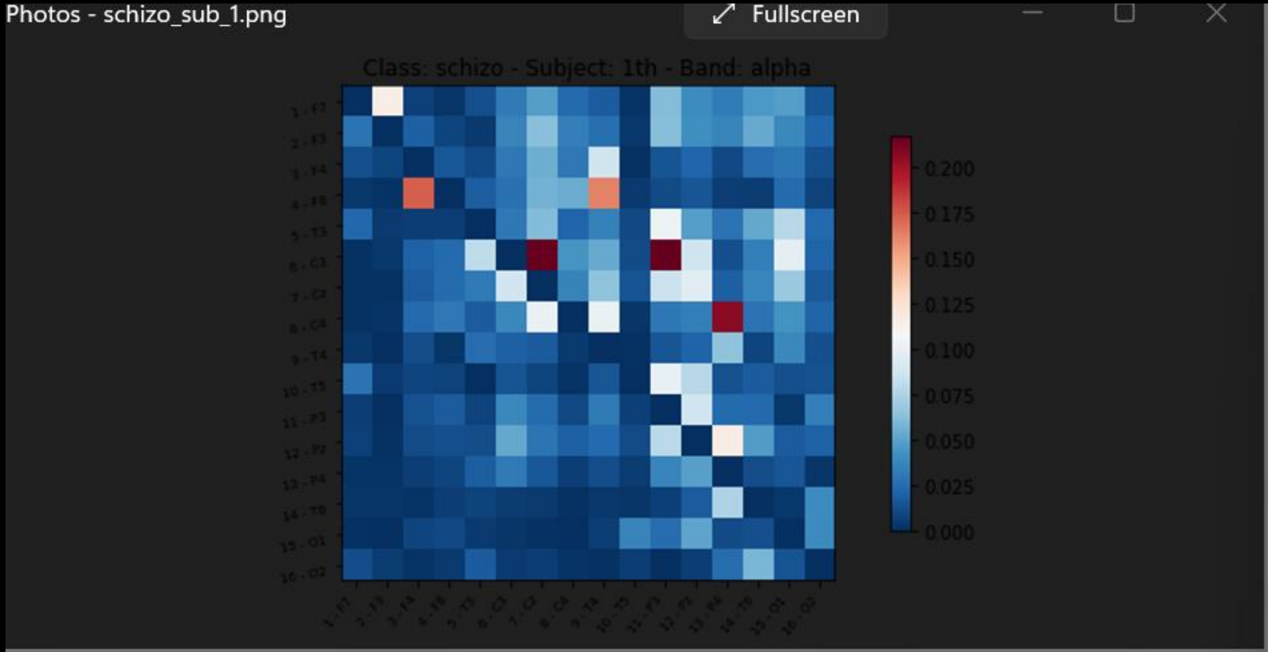
Results

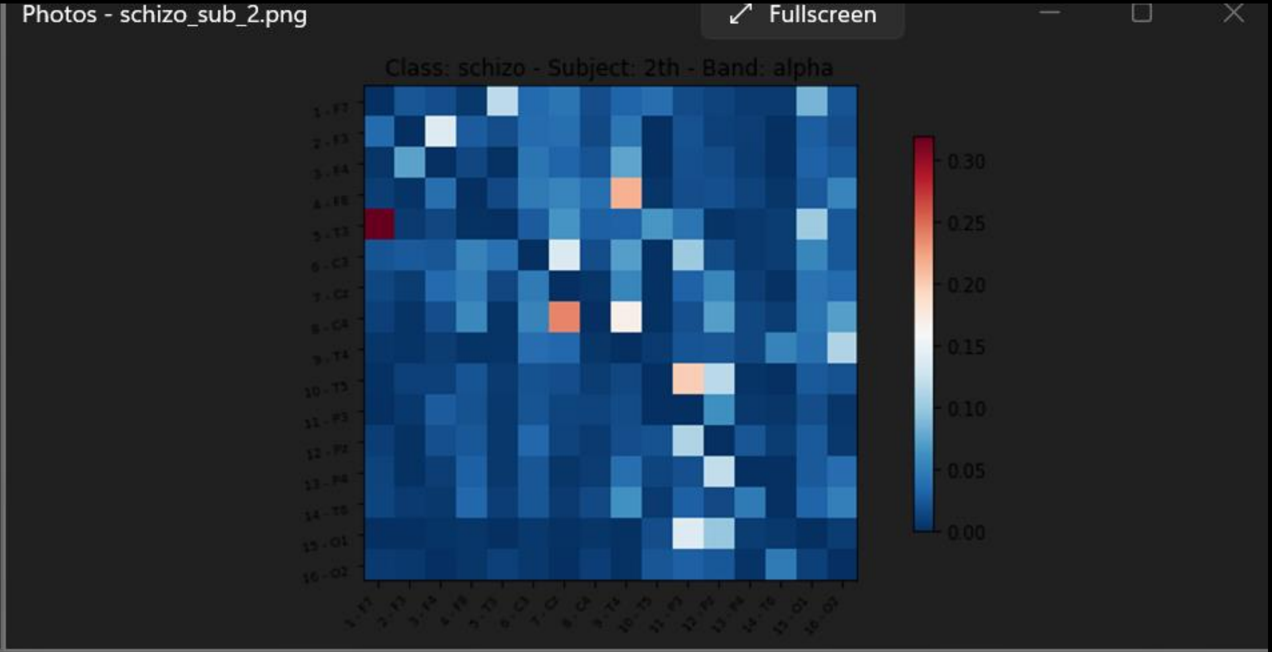

Block15th
PDC : 1→1
PDC : 2→1
PDC : 3→1
PDC : 4→1
PDC : 1→2
PDC : 2→2
PDC : 3→2
PDC : 4→2
PDC : 1→3
PDC : 2→3
PDC : 3→3
PDC : 4→3
PDC : 1→4
PDC : 2→4
PDC : 3→4
PDC : 4→4
1
0.5
0
0
50

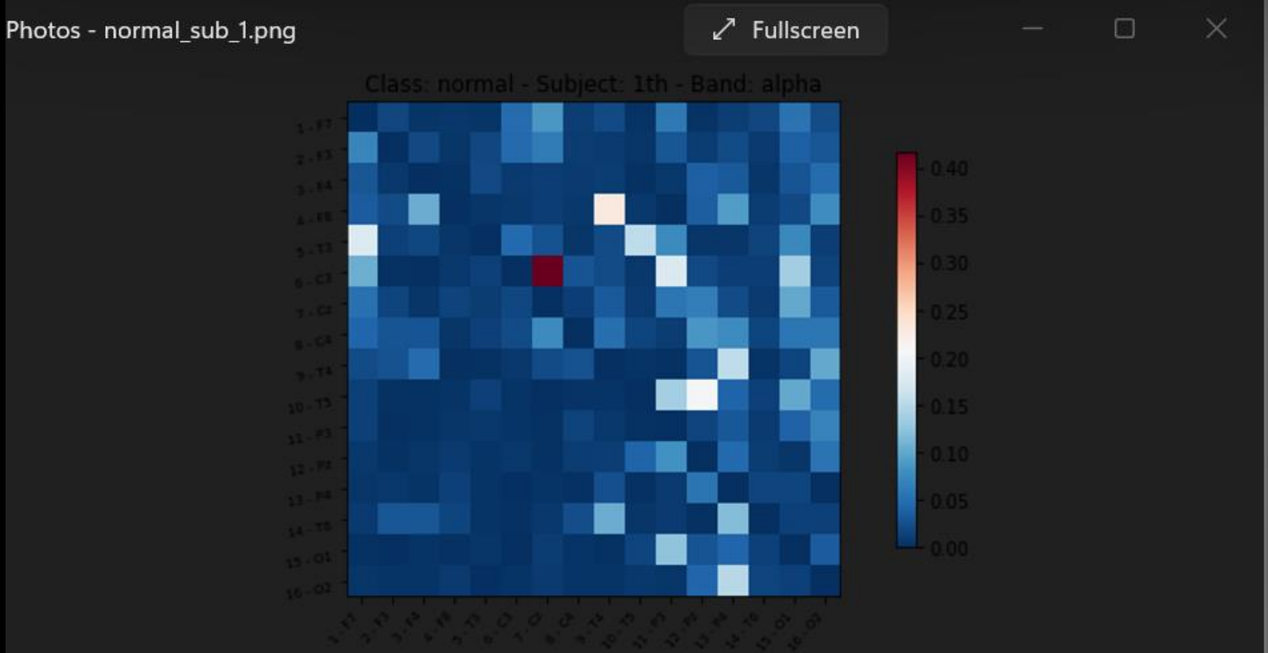
Results

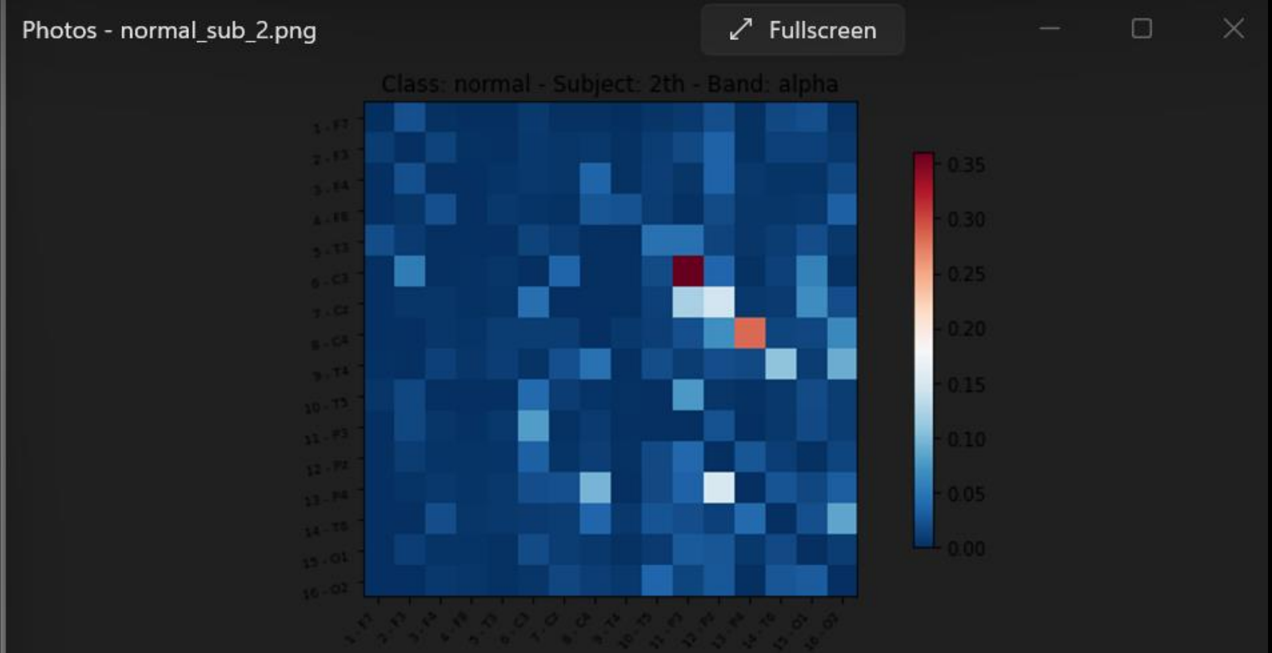

Block16th
PDC : 1→1
PDC : 2→1
PDC : 3→1
PDC : 4→1
PDC : 1→2
PDC : 2→2
PDC : 3→2
PDC : 4→2
PDC : 1→3
PDC : 2→3
PDC : 3→3
PDC : 4→3
PDC : 1→4
PDC : 2→4
PDC : 3→4
PDC : 4→4
0
0.5
1
0
50

Results

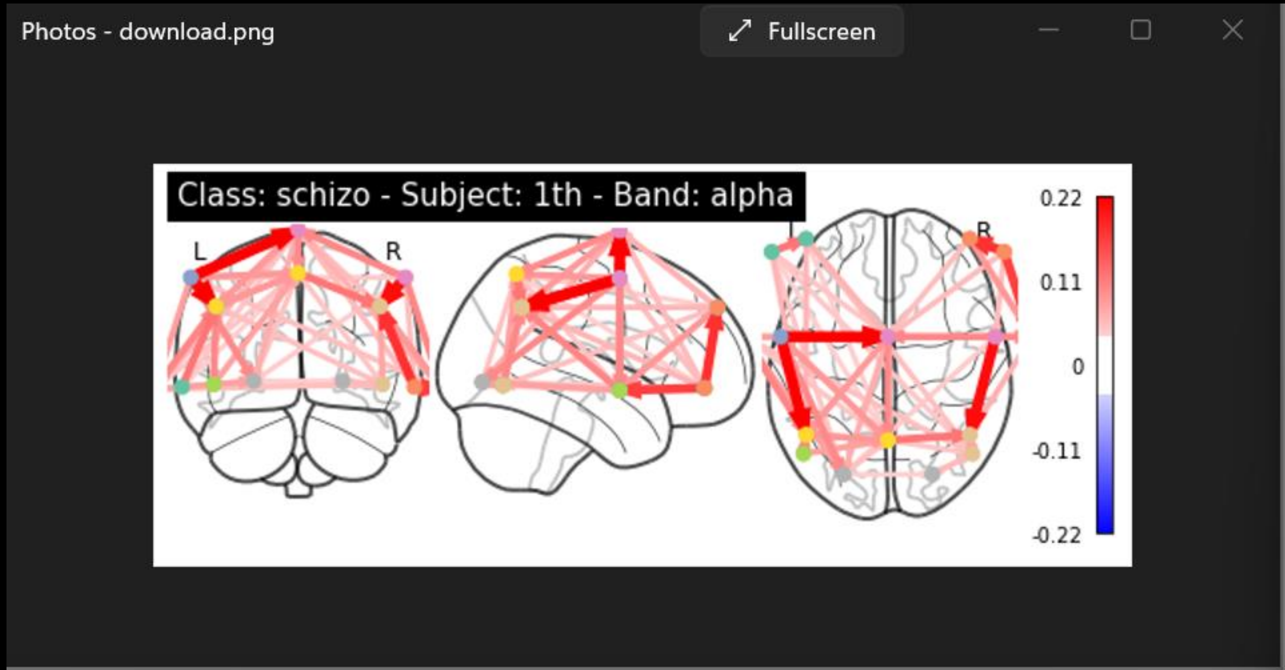

Photos - schizo_sub_1.png
Fullscreen
Class: schizo - Subject: 1th - Band: alpha
0.200
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000

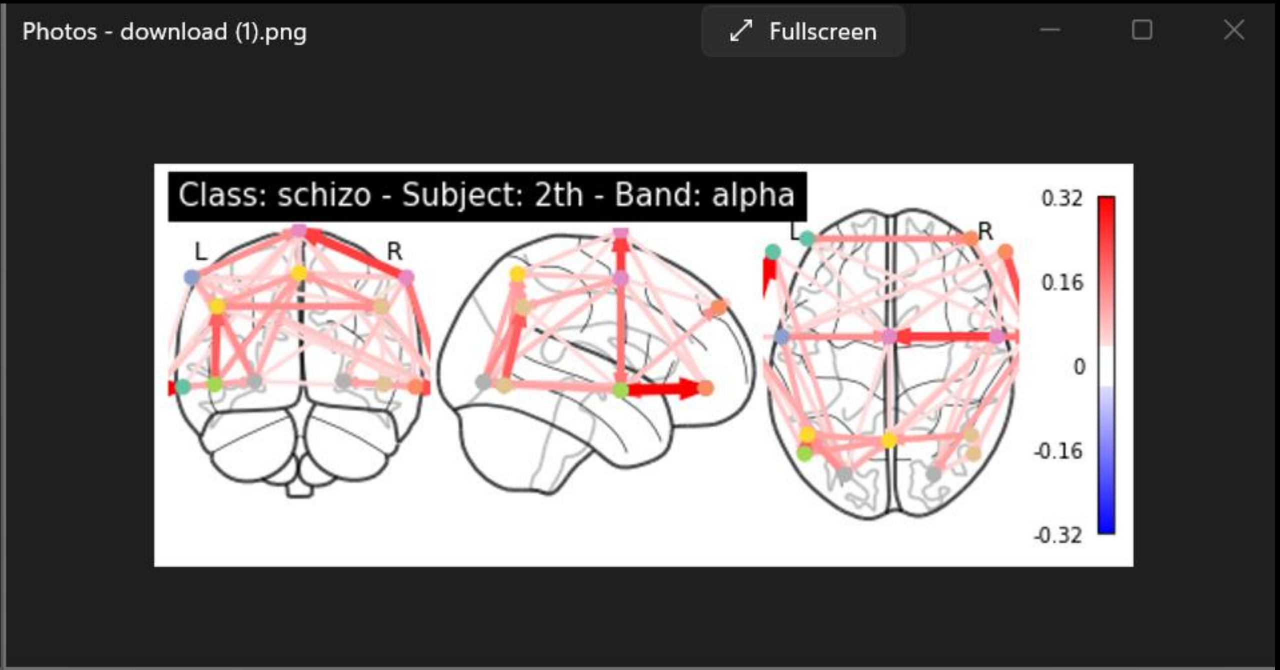

Photos - schizo_sub_2.png
Fullscreen
Class: schizo - Subject: 2th - Band: alpha
0.30
0.25
0.20
0.15
0.10
0.05
0.00

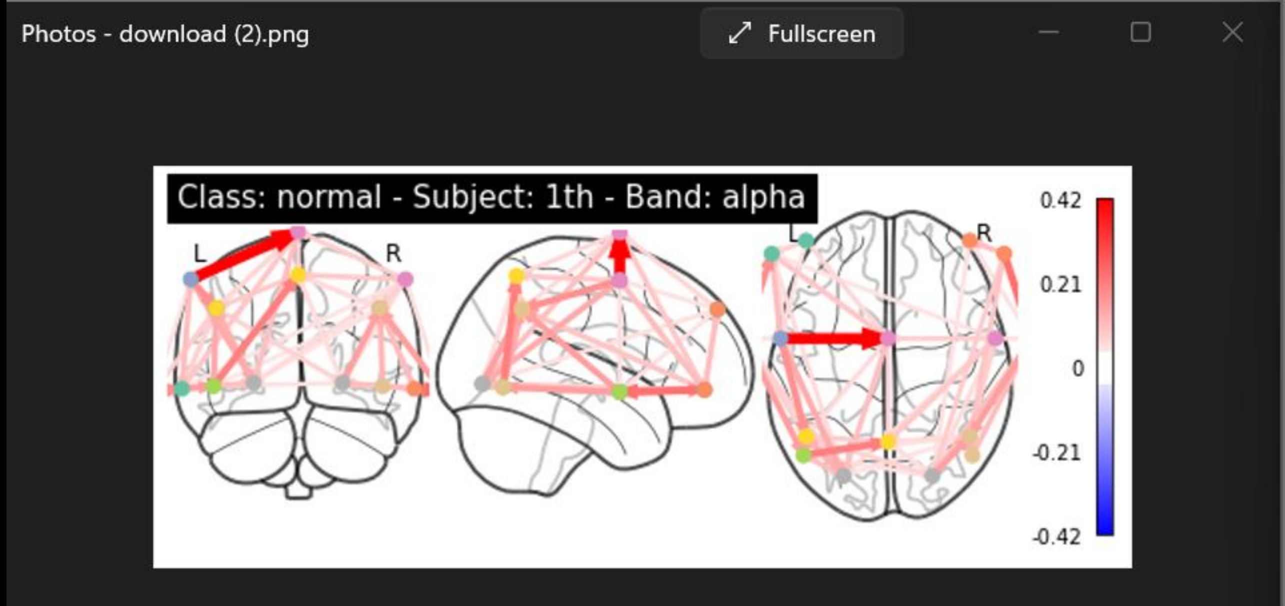

Photos - normal_sub_1.png
Fullscreen
Class: normal - Subject: 1th - Band: alpha
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00

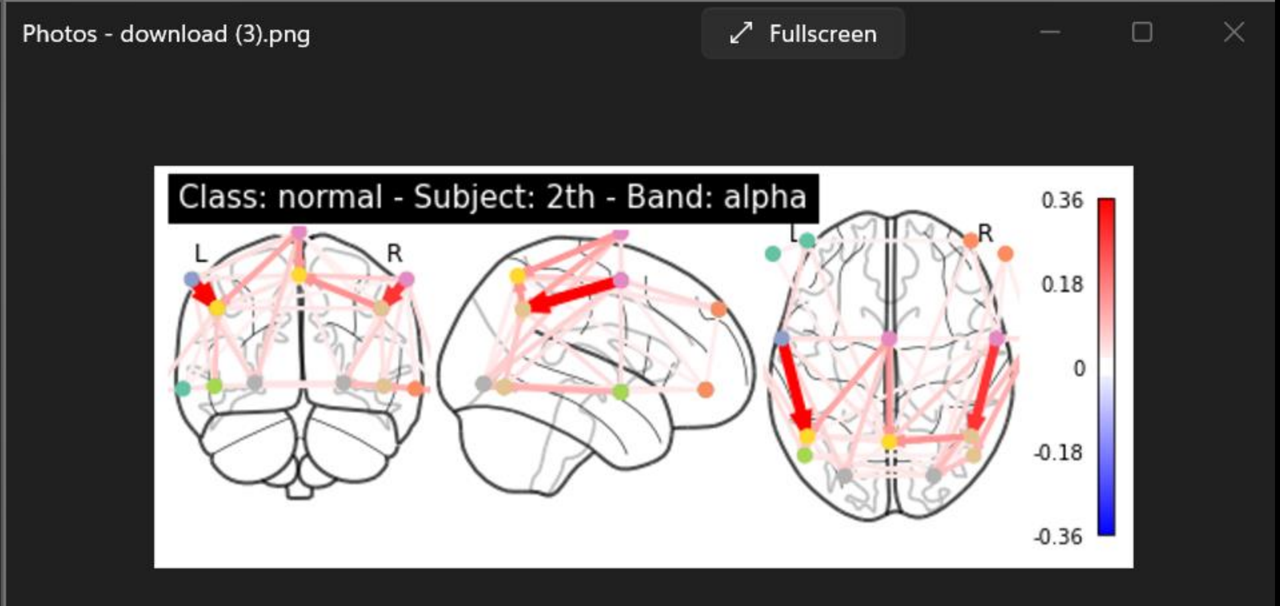

Photos - normal_sub_2.png
Fullscreen
Class: normal - Subject: 2th - Band: alpha
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00

Results


Photos - download.png
Fullscreen
Class: schizo - Subject: 1th - Band: alpha
L
R
0.22
0.11
0
-0.11
-0.22


Photos - download (1).png
Fullscreen
Class: schizo - Subject: 2th - Band: alpha
L
R
0.32
0.16
0
-0.16
-0.32


Photos - download (2).png
Fullscreen
Class: normal - Subject: 1th - Band: alpha
L
R
0.42
0.21
0
-0.21
-0.42


Photos - download (3).png
Fullscreen
Class: normal - Subject: 2th - Band: alpha
L
R
0.36
0.18
0
-0.18
-0.36