Brain network study during resting states

Bioinformatics Project Part 1

Omirbanu NURASSILOVA, Dusica STEPIC, Francesca PORCU [1]

Introduction

The goal of the assignment was to perform a comparative analysis of two datasets containing EEG data[2]. This data was gathered from 64 electrodes with the subject at rest in eyes-open(EO) and eyes-closed(EC) conditions, respectively.

During data analysis, we covered the following topics:

1. Connectivity graphs
2. Graph-theory indices
3. Motif analysis
4. Community detection

Part 1. Connectivity graph

In this part, we were aimed to estimate functional brain connectivity among 64 channels. In order to do that, we used two types of Multivariate Autoregressive models: Partial Directed Coherence (PDC), Direct Transfer Function (DTF). The Multivariate Autoregressive(MAR) model characterizes interregional dependencies within data, specifically in terms of the historical influence one variable has on another[3]. We used python implementation of Multivariate Autoregressive modes[4]. From that implementation, we used methods that perform estimation using PDC, DTF. We were aimed to obtain a network with a density of 20% applying threshold. In order to find a threshold, we wrote a function that calculates it using the formula of the density of directed graphs. Applying that value of the threshold to the output of estimation gives us an adjacency matrix and we can build the target graph. We performed this procedure for both of the given conditions. As shown in figure 1, in both conditions(EO and EC) using both MAR models with 20% density graphs are still connected.

In addition, we had a task where we needed to consider subset with 19 electrodes. We needed again to repeat process that was described above but in addition we were required to filter out values that are not significantly different from 0 ($PDC(i, j) \neq 0$ with $p < 5\%$). In the python implementation there is a method that is written for that. We decided to use 100 repetition of Bootstrap sampling and 0.05 for alpha value in the method. As the result we obtained adjacency matrix that can be used for visualizing a graph. Figure 2 illustrates graphs obtained with 19 channels. In this figure where we are considering first condition we see that we have 3 disconnected nodes (nodes which do not have any incoming or outgoing edges) while in the second condition we have 2 disconnected nodes. They are different in both cases.

We were provided with cartesian coordinates of channels. Using them we made topological representation of the network. We performed it to both variants of the graphs we had with 64 and 19 channels. Topological representation of 64 channels with density 0.05 is shown in figure 3 and for 19 channels it is shown in figure 4. When we are considering 64 channels with density 0.05 in condition when eyes are closed we see less connections in right-side of central lobe in comparison with condition when eyes are opened. And with opened eyes condition there are more outgoing edges from frontal lobe to parietal lobe

---

[1] University of Rome, Sapienza, MSc Data Science, Bioinformatics class 2019/2020, Group No: 8

[2] https://physionet.org/physiobank/database/eegmmidb/

[3] https://www.fil.ion.ucl.ac.uk/~wpenny/publications/spm-book/mar.pdf

[4] https://connectivipy.readthedocs.io/en/latest/tutorial.html

in comparison with closed eyes condition. On the contrary, in closed eyes condition we can see that we have more outgoing edges from parietal lobe to frontal lobe than in opened eyes condition. Let's consider now figure 4. In opened eyes case we see that there are less connections in frontal lobe and on the left side in comparison with closed eyes case.

Part 2. Graph theory indices

In the second part, we were aimed to compute binary global (clustering coefficient, path length) and local (degree, in/out degree) graph indices. In order to perform this task, we started from the DTF graph generated in the previous point. Results of the global graph indices are shown in Table 1, in which we compared the values obtained both in the eyes-open(EO) condition and in the eyes-closed(EC) condition. Then, we observed the global graph indices through PDC estimator (Table 2).

The clustering coefficient, when applied to a single node, is a measure of how complete the neighborhood of a node is. When applied to an entire network, it is the average clustering coefficient over all of the nodes in the network. The closer the clustering coefficient is to 1, the more likely it is for the network to form clusters.

The average path length is defined as the average number of steps along the shortest paths for all possible pairs of network nodes. In a real network, a short average path length facilitates the communication and/or influence between nodes. Observing Table 1 and Table 2, we can notice a low average clustering coefficient and a short average path length in both eyes-open and eyes-closed conditions.

Results of the local graph indices are shown in Table 3 and in Table 4, in which we summarized the top 10 nodes sorted by highest degree in EO and EC condition, respectively. We computed these values using the DTF estimator. The degree of a node is the total number of connections with other vertices. The greater the degree, the more important is the presence of that node for the whole system. In a directed graph, as in our case, the degree can be split into in-degree and out-degree. In-degree of a node is the total amount of incoming links; instead, out-degree of a node is the total amount of outgoing links. Observing the tables related to the local indices, we can notice that the degree distribution when the subject has his eyes open is pretty much similar to the other case.

Then, performing another task, we plot a topographical representation of local indices using cartesian coordinates of EEG channels (from Figure 5 to 7). For a better comprehension, we highlighted in blue the 10 highest degree nodes. We can observe that, in both cases, the nodes with high in-degree values are homogeneously located in the different brain lobes, with a slight tendency to the right hemisphere; instead, the highest out-degree nodes are mostly located in the frontal/parietal lobe, with a high tendency to the left hemisphere.

In addition, we tried to compute the small-world organization of our network. A graph has a small-world property if it is characterized by a high clustering coefficient and a short path length. The main mechanism to construct small-world networks is the Watts-Strogatz model[5]. We used this model to compute a small-world network with same number of nodes and same average degree of our directed graph. Small-world properties are found in many real-world phenomena, including biological and neural networks.

We started from a regular ring lattice graph with N vertices and k edges per vertex; then, we rewire each edge at random with probability β. The Watts-Strogatz model builds an undirected graph, so we used to_directed() function of NetworkX Python package to return a directed representation of the graph, as in our case. We built three graphs with probability β equal to 0, 0.25 and 1, respectively; both for EO and EC subject-condition (Figure 8 and 9). When β=0, we obtain a completely regular lattice graph, with no rewiring and each node connected to k of its neighbors. If no edge is rewired, we expect that the distances between each pair of nodes would be long. Also, if k is large enough, the ring lattice starts to form many triangles. So, it is characterized by a high clustering coefficient and a long path length. With β=1, we rewire every

[5] https://www.nature.com/articles/30918

single edge; so, we obtain a completely random graph with short path length and small clustering coefficient. With β=0.25, we obtain a small-world network. Some edges are rewired; so, the distances between nodes are reduced but a high clustering coefficient is maintained. Hence, we can notice that the average clustering coefficient and the shortest path of a small world network depend on the parameters k and β[6].

We tried to quantify network small-worldness by a small-coefficient, σ, calculated by comparing clustering and path length of a given network to an equivalent (directed) random network with same degree on average[7].

If σ coefficient is greater than 1, the network presents small-world properties. Results of the σ coefficient are shown in Table 5, in which we compared the values obtained both in EO and EC conditions using the DTF estimator. Observing our table, we can notice that σ coefficient values are greater than 1. We obtained a small-world network in both cases. However, this metric is known to perform poorly because it is heavily influenced by the network's size[7].

Part 3. Motif analysis

In this part we are aimed to analyse motifs presence in the network. Network motifs are sub-graphs that repeat themselves in a specific network. Each of these sub-graphs, defined by a particular pattern of interactions between vertices, may reflect a framework in which particular functions are achieved efficiently. Indeed, motifs are of notable importance largely because they may reflect functional properties.[8] We considered configurations which contain 3 nodes. There are possible 13 configurations. We implemented our function that counts their frequencies in python. Additionally, we needed to find their significance. In order to do that we needed to consider also random graph with the same number of nodes and density. We performed Monte Carlo simulations to get mean and standard deviation of frequencies of configurations in random graphs. We needed these two values in order to count Z values of configurations. The larger the Z value then its significance is larger also. Results of counting frequencies and comparing with random graphs are shown in Table 6. Among 13 configurations 7 have high Z-value in condition where eyes are opened. Especially, Z-value of configuration when 3 nodes construct a triangle where they go consequently in both directions even though it has really small frequency in both cases. Configuration 'A->B<-C' has the highest frequency and quite high Z-value in both cases. Configurations that have Z-value less than -0.9 can be considered as anti-motifs. From first condition we have 3 anti-motifs. Two of them are also anti-motifs for second configuration however one of them is not the same and it is configuration that was not nor motif or anti-motif in first condition.

We performed also topological representation of network considering only configuration 'A->B<-C' . In order to perform this task we wrote a procedure in python that takes into consideration this configuration. Since it has the highest frequency it covers all the nodes. In order to see the difference visually we used graphs with density 0.05. Graphs are represented in Figure 10. When eyes are closed there are few edges in nodes in right central lobe part and node 'P7' has more connections with frontal lobe whereas with opened eyes it has less connections. And with opened eyes node 'Af7' has more connections than it has in closed eyes case.

In parieto-occipital scalp region we selected 2 channels to consider and show in which motifs they are involved. They are 'Po8', 'Poz'. For the first one we have four configurations in eyes opened and eight in case when eyes are closed. And for the second one: six and nine in opened and closed eyes cases respectively. Motifs related to node 'Po8' are shown in figures 11 and 12. Motifs related to node 'Poz' are shown in figures 13 and 14.

[6] https://www.coursera.org/lecture/python-social-network-analysis/small-world-networks-Iv4e8
[7] https://en.wikipedia.org/wiki/Small-world_network
[8] https://en.wikipedia.org/wiki/Network_motif

Part 4. Community detection

A brain still remains a mystery that needs to be unraveled. Over the years, a lot of research has been done in order to understand this powerful organ better and have a clearer picture of the way it works and how certain brain regions interact with each other and share information flow. Many algorithms are developed for understanding brain connectivity and information flows using graph theory. However, we will see here how Louvain[9], Leicht & Newman[10], and Infomap[11] algorithms behave with two analyzed datasets containing EEG data.

The main research question regarding this part is making inference by seeing how different parts of brains can be grouped by using different techniques that are information-based(Infomap algorithm) or based on the modularity of the network(Louvain and Leicht & Newman algorithm). Clustering algorithms seek to capture the intuitive notion that nodes should be connected to many nodes in the same community (intra-cluster density) but connected to few nodes in other communities (inter-cluster sparsity).[12]

Figures 15, 16, 17 show community structures as mentioned in the previous sections, for two different conditions: eyes-closed (EC), and eyes-opened (EO). Colored vertices correspond to the identified communities and their location according to the channel labels. Tables 7 and 8 show the composition of the communities obtained by Louvain algorithm. By observing the results of the modularity-based algorithms, we see that communities in the results of the Louvain algorithm in both states look quite differentiable. Leicht-Newman Algorithm gave similar results as the Louvain algorithm. Moreover, the infomap algorithm created one community in the EC state. While, in the EO state, a lot of nodes from frontal, right temporal, parietal and occipital parts are labeled as part of the bigger community(region) of the brain. While comparing the execution times of the algorithms, Louvain was the most efficient, while Leicht & Newman algorithm had worse time performance.

Figures and Tables

Task	***Class***
1.1	*mandatory*
1.2	A
1.4	*D*
1.5	*C*
2.1	*mandatory*
2.2	*D*
2.3	*B*
2.5	*B*
3.1	*mandatory*
3.2	*C*
3.3	*C*
4.1	*mandatory*
4.2	*B*
4.3	*C*

Table 0 - The list of tasks chosen for the project

[9] https://louvain-igraph.readthedocs.io/en/latest/
[10] www.mapequation.org
[11] www.mapequation.org
[12] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4938516/

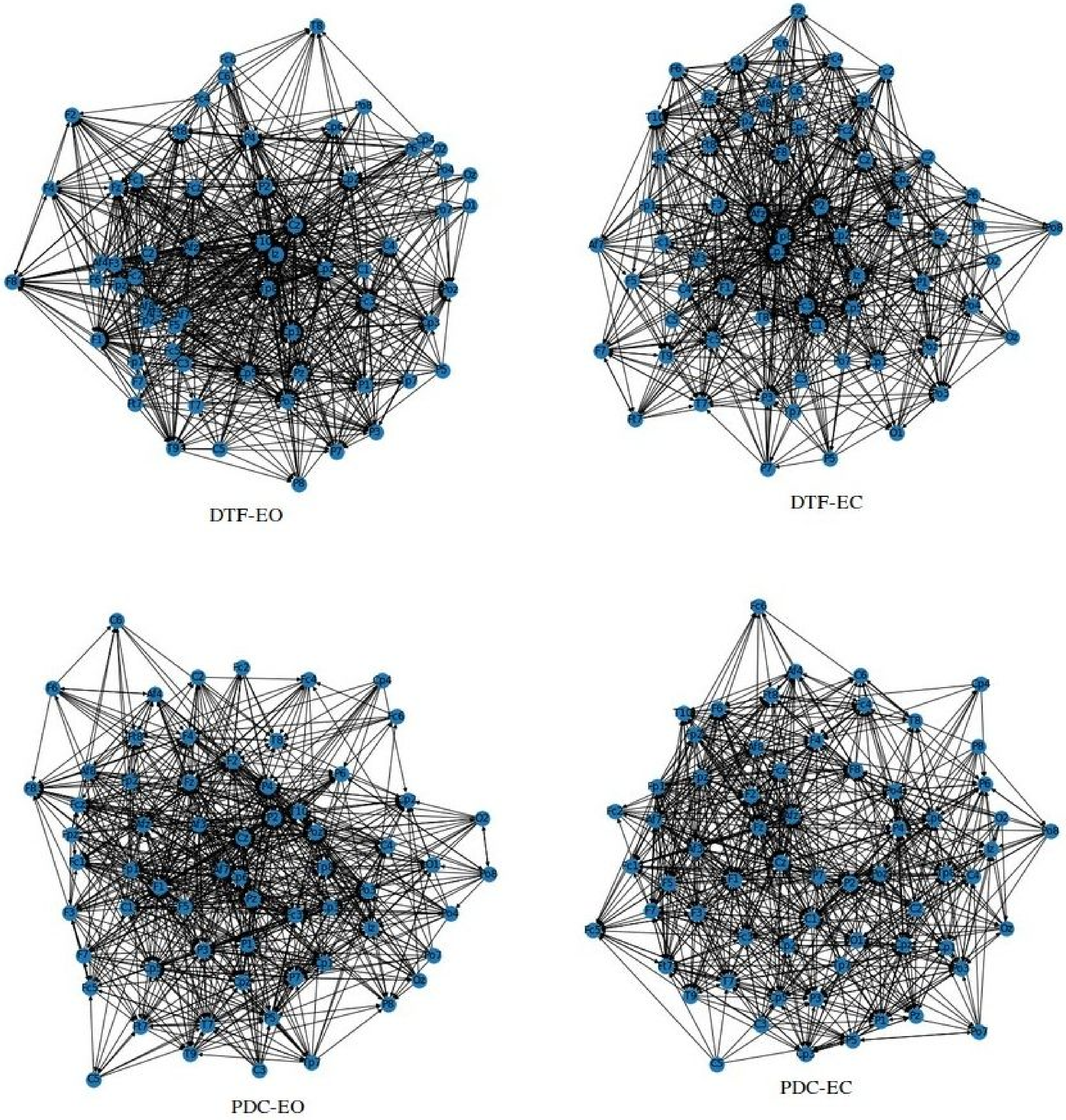


Figure 1 - Network with a density 20% applying threshold

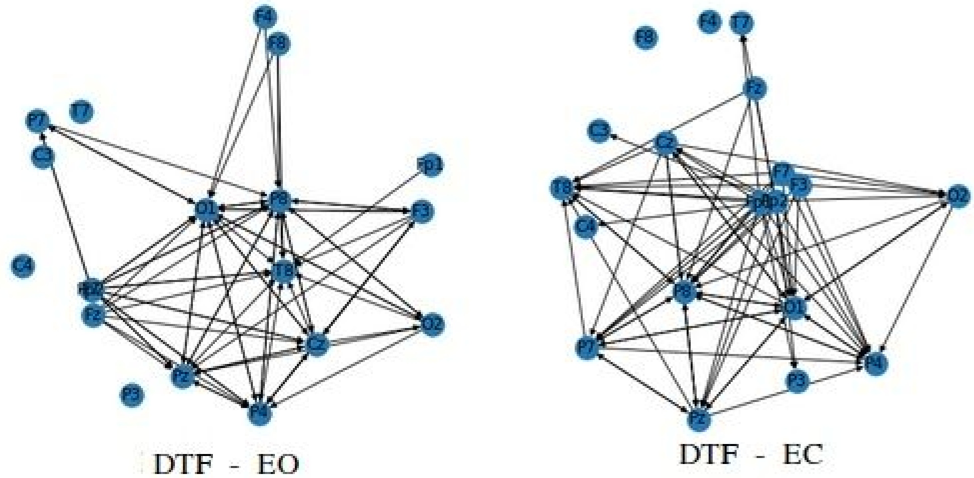


Figure 2 - DTF of 19 channels

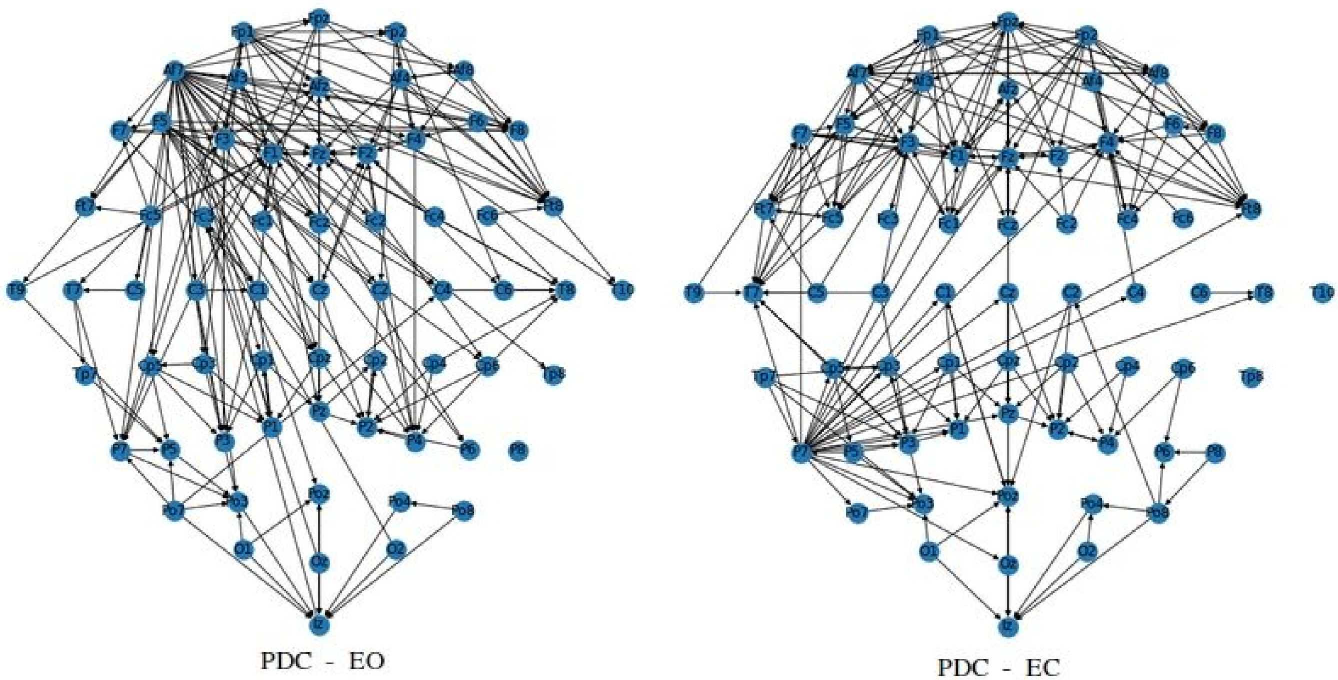


Figure 3 - PDC with density 0.05 and cartesian coordinates

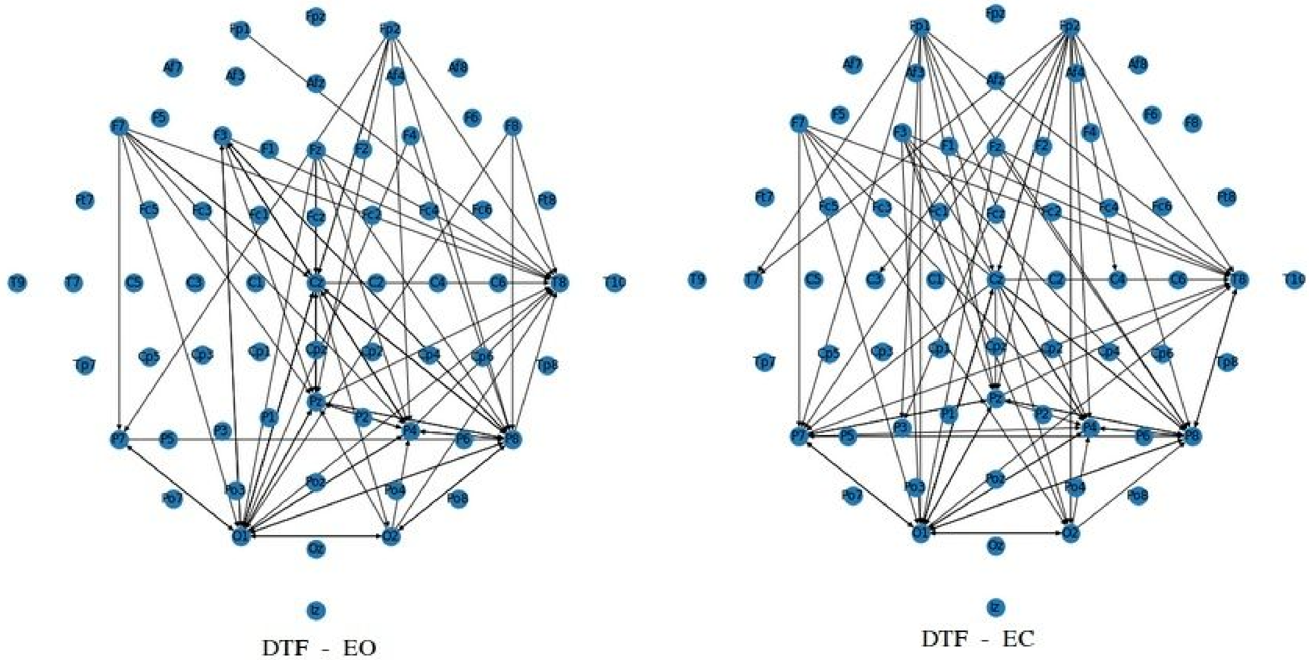


Figure 4 - DTF with density 0.2 and cartesian coordinates for 19 channels

Condition	Index	DTF
Eyes-open	Average clustering coefficient	0.41
Eyes-open	Average path length	0.84
Eyes-closed	Average clustering coefficient	0.41
Eyes-closed	Average path length	1.56

Table 1 - DTF global indices

Condition	Index	PDC
Eyes-open	Average clustering coefficient	0.33
Eyes-open	Average path length	2.67
Eyes-closed	Average clustering coefficient	0.36
Eyes-closed	Average path length	1.98

Table 2 - PDC global indices

Node	In-degree
T10	60
Iz	60
Tp8	56
Cz	47
P2	38
Cp2	33
Cp1	32
Fz	30
Fc3	29
Afz	29

Node	Out-degree
Cp5	21
Afz	20
Cp3	19
Af7	19
Fc5	18
Fc3	18
Fp1	18
Fp2	18
Af3	18
F5	18

Node	Degree
Iz	68
Cz	61
Tp8	61
T10	60
Cp5	49
Cp1	49
Afz	49
Fc3	47
P2	46
Cp2	44

Table 3 - Local indices for DTF eyes-open (EO)

Node	In-degree
Afz	53
Cp3	49
P2	48
Tp8	46
Fc3	38
C1	32
F1	31
Iz	28
T10	26
Cp2	25

Node	Out-degree
Cp3	19
Fpz	19
Af3	19
Pz	19
Fc5	18
Fp1	18
Afz	18
P1	17
Cp5	16
Af7	16

Node	Degree
Afz	71
Cp3	68
P2	62
Tp8	52
Fc3	51
C1	46
Iz	44
Cp2	40
F1	39
F3	37

Table 4 - Local indices for DTF eyes-closed (EC)

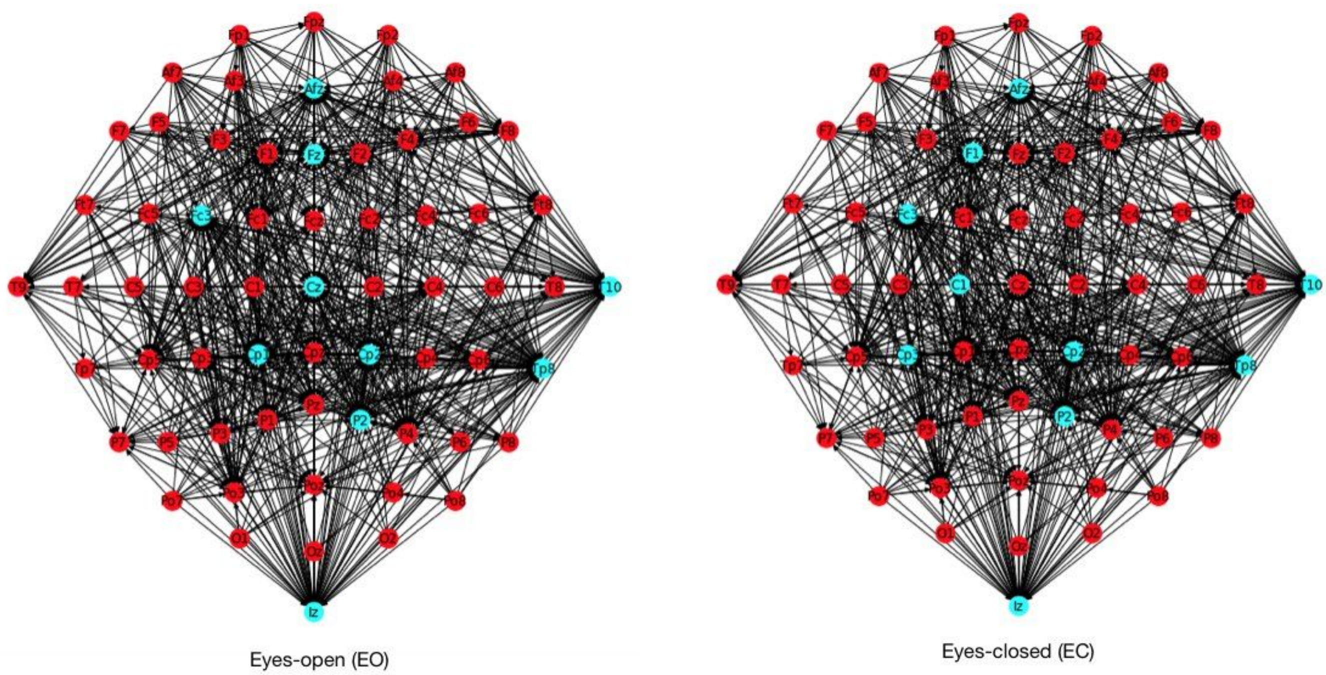
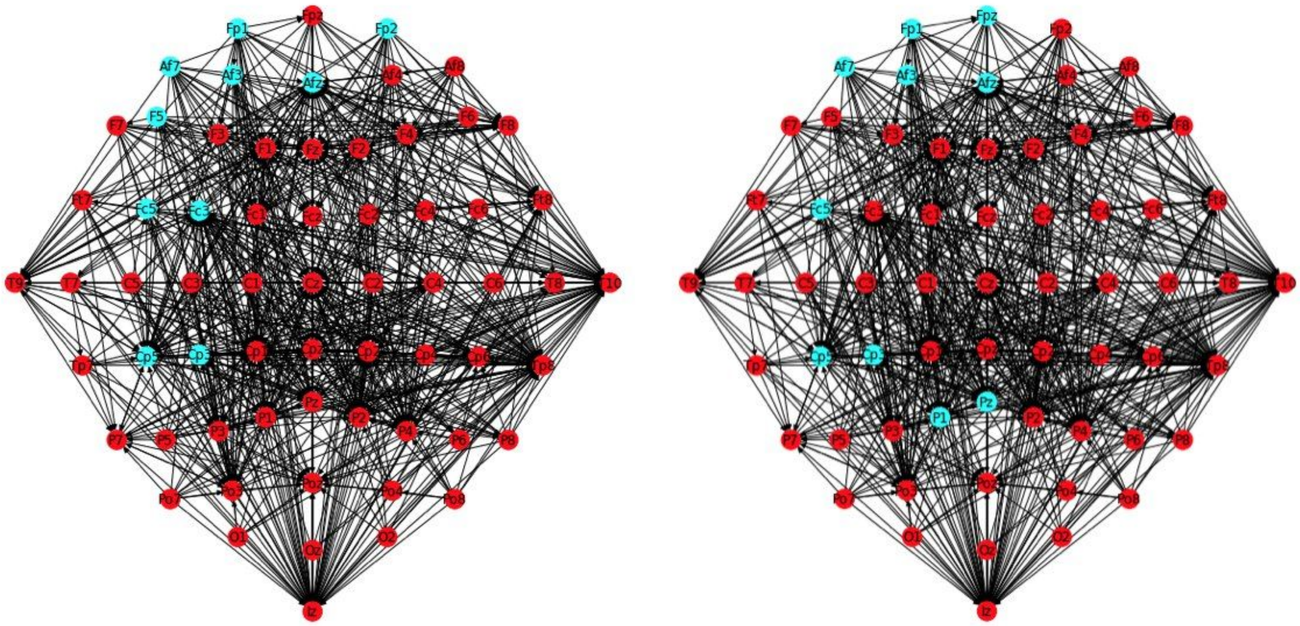
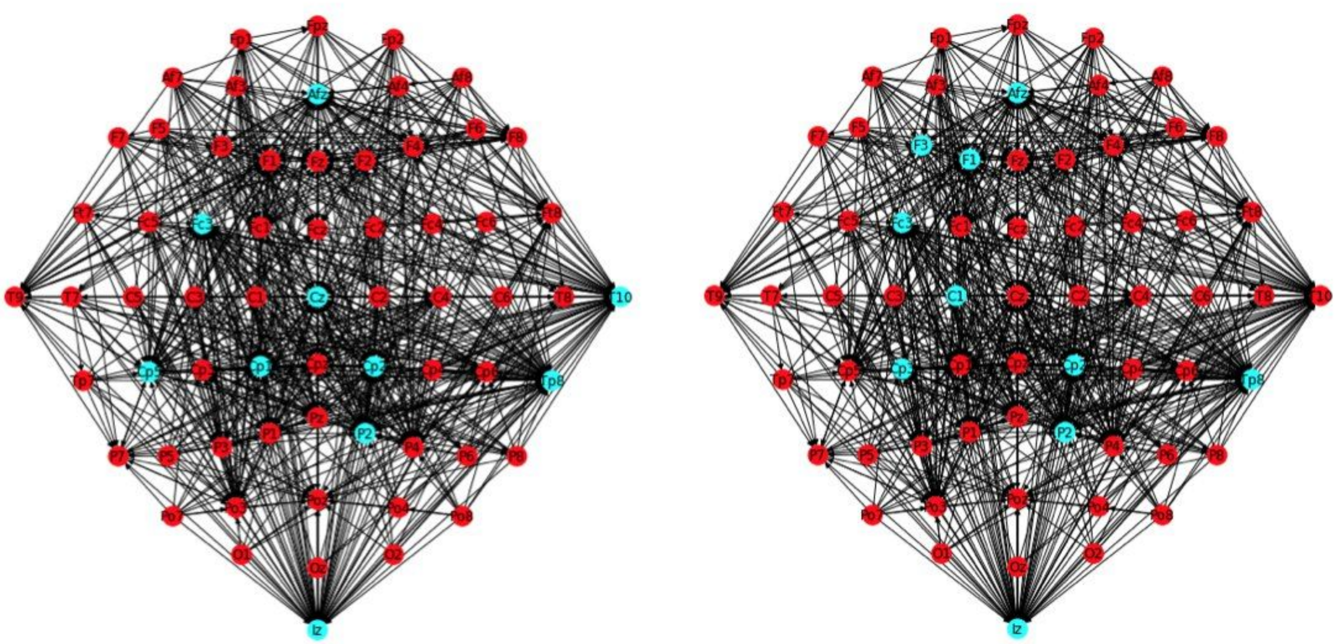


Figure 5 - DTF In-degree



Eyes-open (EO) Eyes-closed (EC)

Figure 6 - DTF Out-degree



Eyes-open (EO) Eyes-closed (EC)

Figure 7 - DTF Degree

```
Directed graph: True
Directed graph: True
Directed graph: True
```

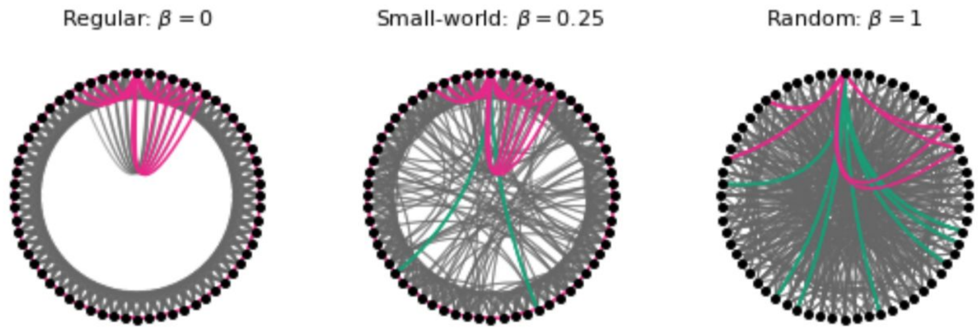


Figure 8 - DTF Small-world network eyes-open (EO)

```
Directed graph: True
Directed graph: True
Directed graph: True
```

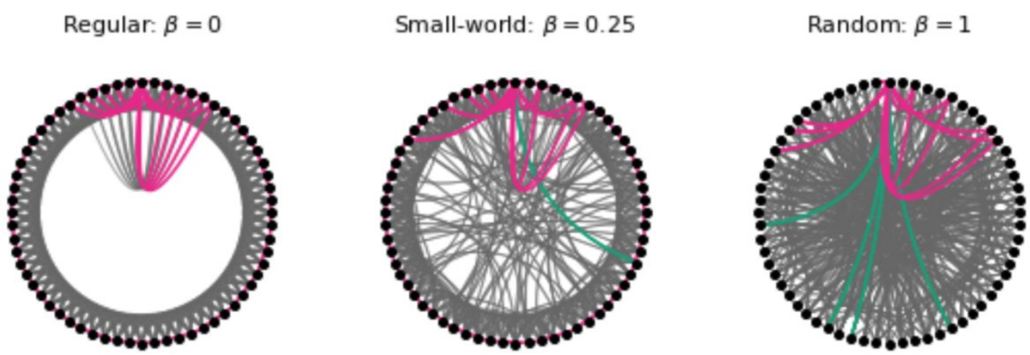


Figure 9 - DTF Small-world network eyes-closed (EC)

Condition	Sigma coefficient
Eyes-open	3.82
Eyes-closed	2.11

Table 5 - Sigma coefficient small-world network

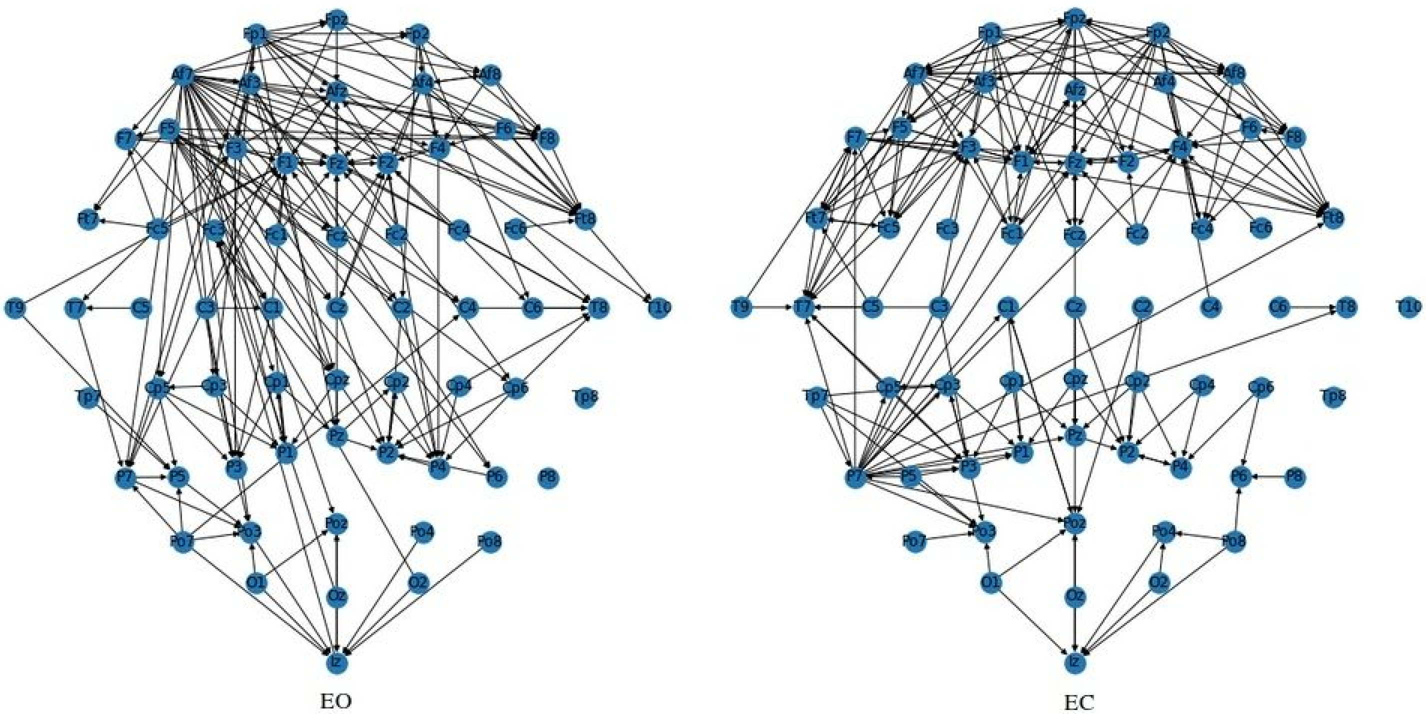


Figure 10 - graph for task 3.2

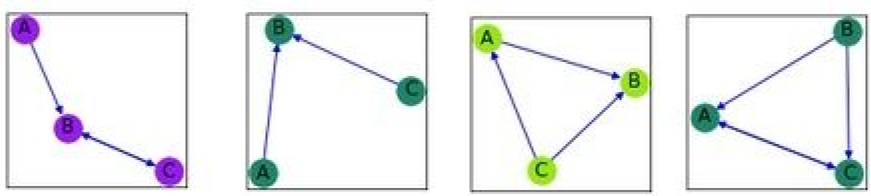


Figure 11 - Po8 EO

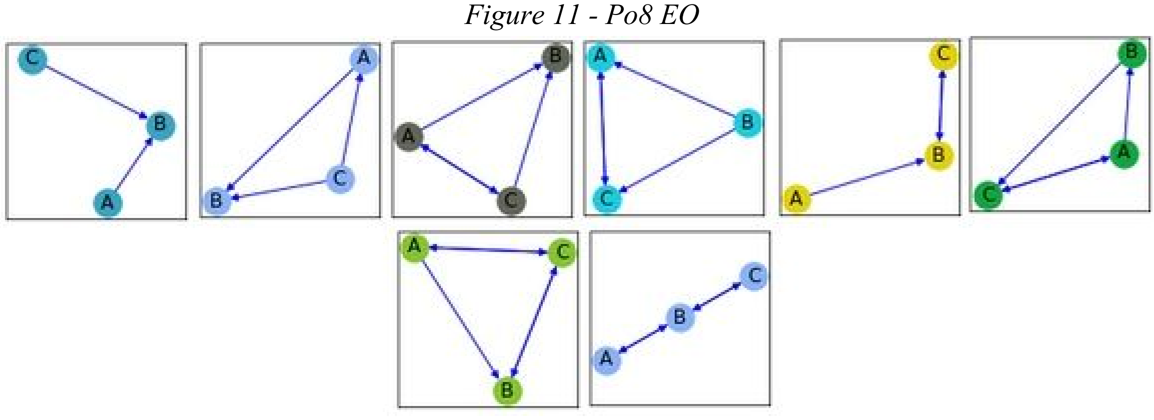


Figure 12 - Po8 EC

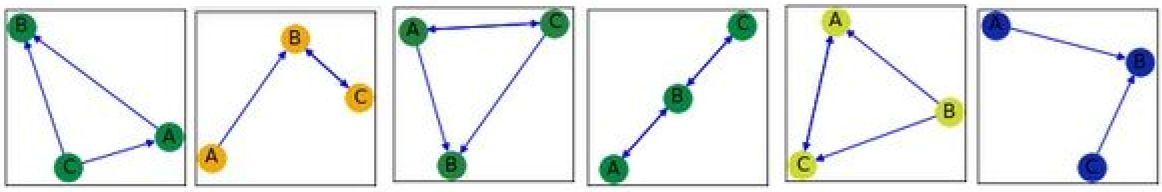


Figure 13 - Poz EO

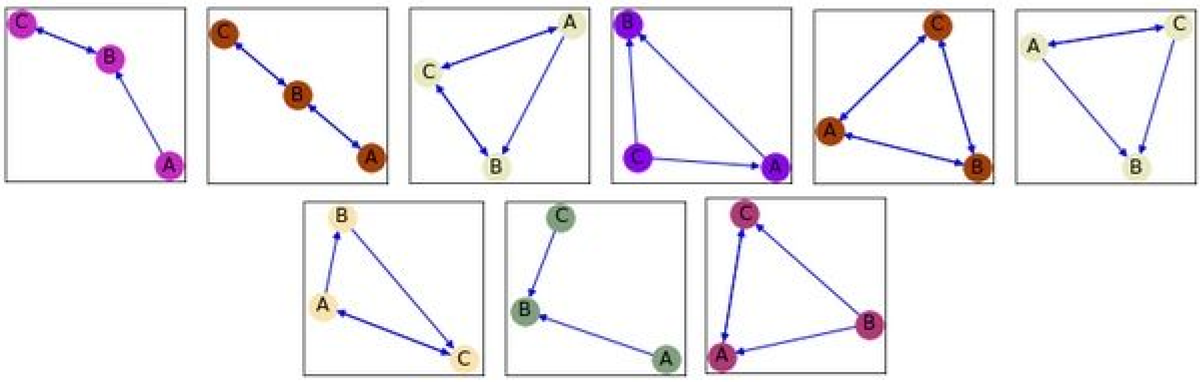
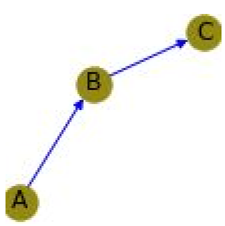
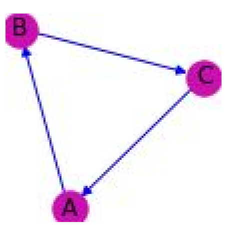
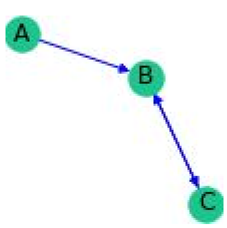
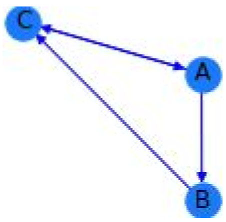
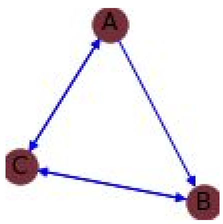
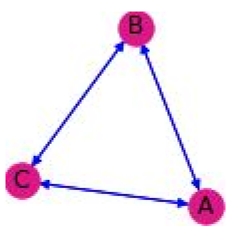
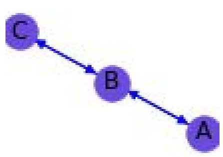
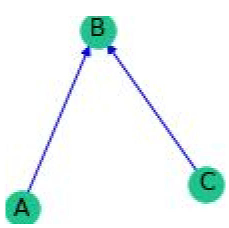
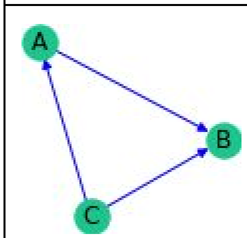


Figure 14 - Poz EC

Configurations	Eyes opened		Eyes closed	
	Frequency	Z	Frequency	Z
	6536	-3.80644247489997	7068	-2.62168772313567
	376	-4.41664112717246	651	-0.2593886089667
	2631	5.49514720404232	3356	9.41470841596516

	371	0.979132578939052	616	6.20871074985355
	3883	-0.6780542742109	7259	3.71279624717091
	58	28.6874815153615	72	35.9362482885425
	355	3.83159388865232	578	9.23608970241209
	11879	23.7103880474756	9278	14.9769223764195
	3660	14.0292706888836	3123	10.356379

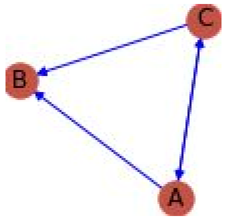
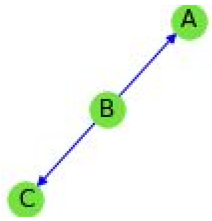
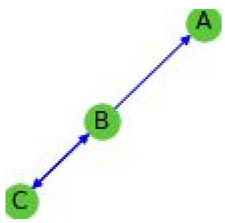
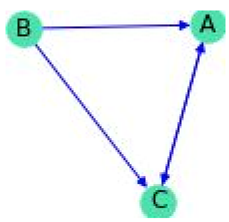
	516	10.0291885168779	649	14.2186990230576
	4812	-0.034302642335016	4475	-1.15587712567501
	900	-3.89593853200326	1367	-1.35195264274215
	1040	25.1990988801795	1191	29.7143013359348

Table 6 - Frequencies and Z-values of motif in networks

Eyes-opened	
Size	**Composition of communities**
22	['Fc5', 'Fc3', 'C5', 'C3', 'C1', 'C4', 'Cp5', 'Cp3', 'Cp1', 'F7', 'F5', 'Ft7', 'T7', 'T9', 'Tp7', 'P7', 'P5', 'P3', 'P1', 'Pz', 'P8', 'Po3']
20	['Fc1', 'Fcz', 'Fc2', 'Fc4', 'Fp1', 'Fpz', 'Fp2', 'Af7', 'Af3', 'Afz', 'Af4', 'Af8', 'F3', 'F1', 'Fz', 'F2', 'F4', 'F6', 'F8', 'Ft8']
22	['Fc6', 'Cz', 'C2', 'C6', 'Cpz', 'Cp2', 'Cp4', 'Cp6', 'T8', 'T10', 'Tp8', 'P2', 'P4', 'P6', 'Po7', 'Poz', 'Po4', 'Po8', 'O1', 'Oz', 'O2', 'Iz']

Table 7 - Composition of communities Louvain algorithm (EO state)

Eyes-closed	
Size	**Composition of communities**
16	['Fc5', 'Fc3', 'C5', 'C3', 'Af3', 'F7', 'F5', 'F3', 'F1', 'Ft7', 'T7', 'T9', 'Tp7', 'Tp8', 'P7', 'P3']
13	['Fc1', 'C1', 'Cz', 'C2', 'C4', 'Cp5', 'Cp3', 'Cp1', 'Cpz', 'Cp2', 'Cp4', 'P2', 'P4']
21	['Fcz', 'Fc2', 'Fc4', 'Fc6', 'C6', 'Cp6', 'Fp1', 'Fpz', 'Fp2', 'Af7', 'Afz', 'Af4', 'Af8', 'Fz', 'F2', 'F4', 'F6', 'F8', 'Ft8', 'T8', 'T10']
14	['P5', 'P1', 'Pz', 'P6', 'P8', 'Po7', 'Po3', 'Poz', 'Po4', 'Po8', 'O1', 'Oz', 'O2', 'Iz']

Table 8 - Composition of communities Louvain algorithm (EC state)

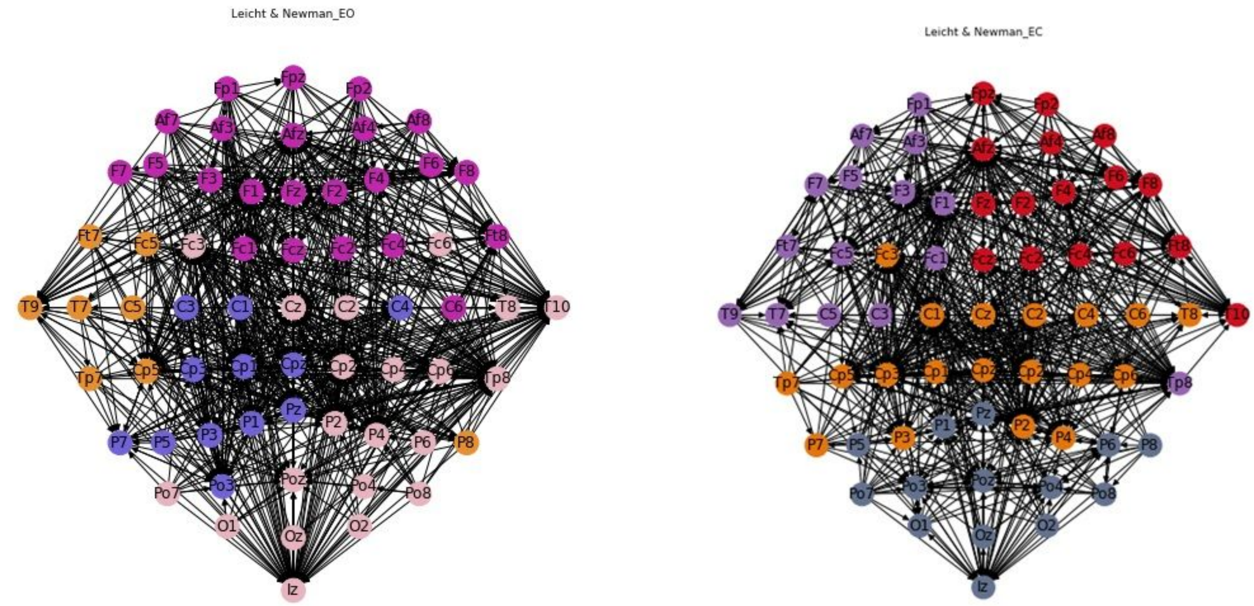


Figure 15 - Communities detected by Leicht-Newman Algorithm

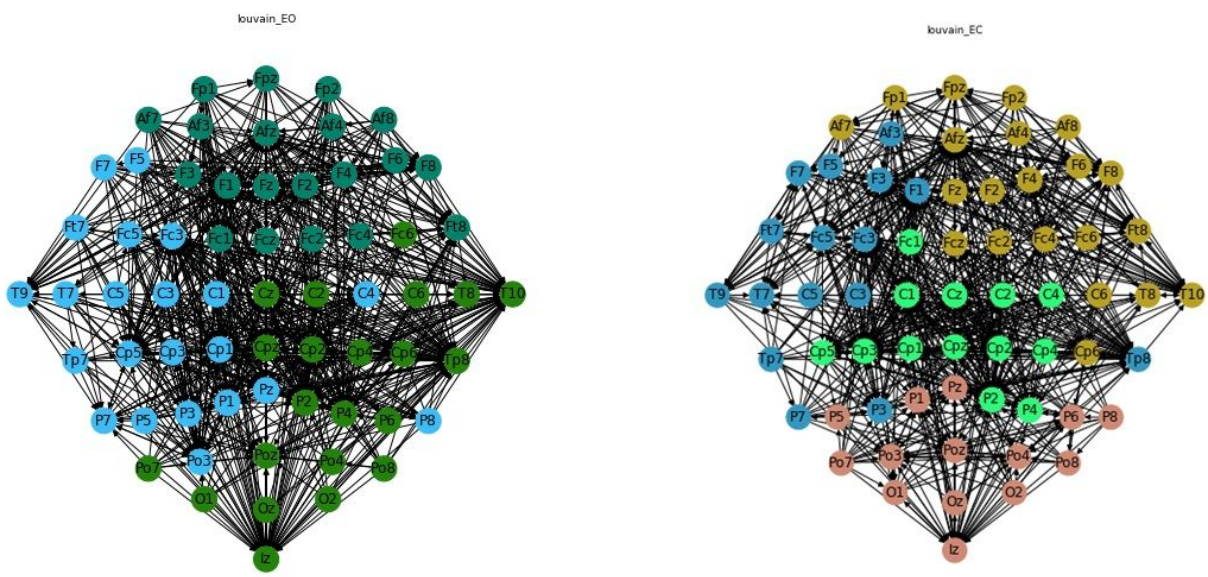


Figure 16 - Communities detected by Louvain Algorithm

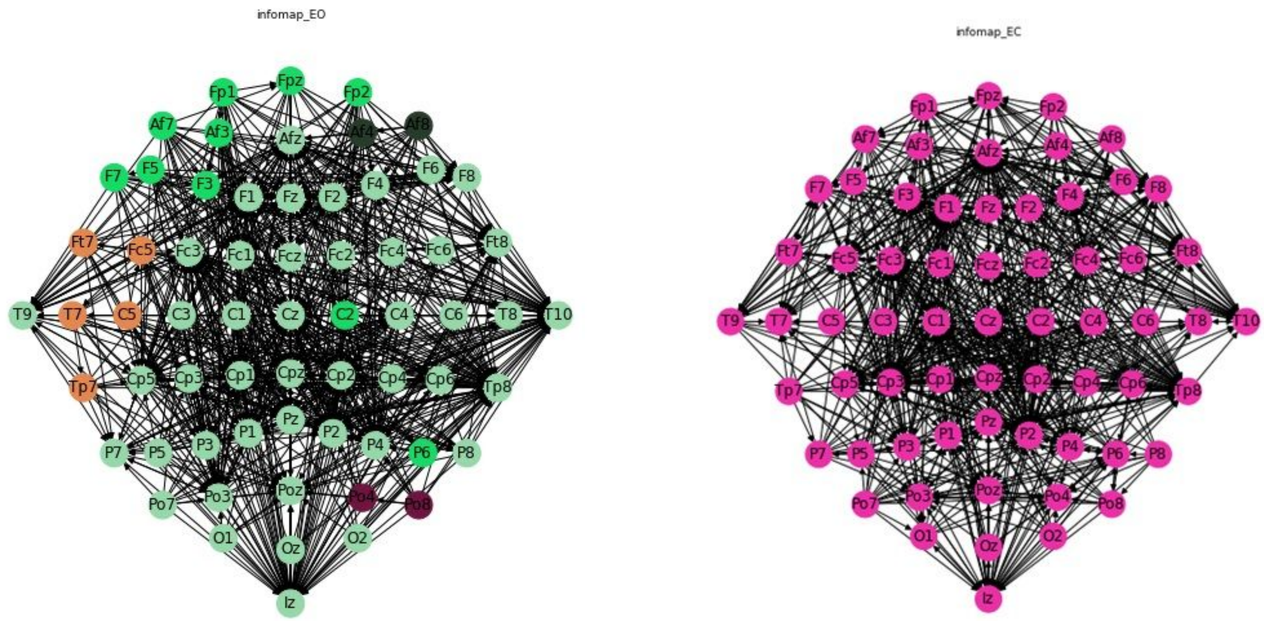


Figure 17 - Communities detected by Infomap Algorithm