UNIVERSITY OF COPENHAGEN
Faculty of Science

Master Thesis

Fayi Zhang

Statistical Inference in Functional Networks

A Statistical Approach with Application in Neuroscience

Supervisor: Susanne Ditlevsen

Submitted on: June 6, 2017

Abstract

The brain is a complex network of connected components, whose interactions evolve dynamically to cooperatively perform specific functions. Building the functional network can help us understand how our brain works and conduct certain tasks. There are many methods that can help build the network. In this project, I will introduce a statistically principled approach to build the functional network.

This analysis is based on the data recorded from 64 sensors during a repeated behavior task. In this project, I will first simulate EEG data with different signal-to-noise ratios (SNRs), correlation ratios and show that this method successfully identifies functional networks and edge densities of confidence for these data. After that, I will employ a principled technique to establish functional networks based on predetermined regions of interest using canonical correlation and analyze the dynamic functional networks associated with it. Finally, I will apply the use of these methods on the real data and build the functional networks for the visual cortex and the whole brain, I will also analyze how the functional networks differ for different stimulus.

Preface

Tempus fugit. This project marks the end of my study at the University of Copenhagen. I have deep appreciation for this university, for the people I met. My experiences here greatly opened my eyes and prepared myself for the next stage of my life.

My deepest gratitude goes first to professor Susanne Ditlevsen, my supervisor, who introduce this interesting topic to me. And I couldn't finish this project without her encouragement and guidance. And I also want to thank Jeppe Hoy Christensen and the department of psychology of the University of Copenhagen for providing me with the data. Finally I want to thank my friends Zhipeng Duan, Yongsong Tian, Yunfeng Liu and Qingchun Song. Thanks for their accompany for these two years in Copenhagen.

Copenhagen, 2017
Fayi Zhang

Contents

Chapter 1

Introduction

Since the nineteenth century, we have realized that the neuronal elements of our brain constitute a very complicated structural network. Many attempts to decipher our brain came up over the past years, building the functional network is one of them. Functional network is built by finding the systematic associations between neural activities in different positions in our brain (different brain regions of interest or recording sites). There are many data types that can be used to estimate the functional network, here, I focus on functional network estimated from EEG data. Due to its excellent performance in inferring dynamic networks, building functional network using EEG data has received more and more attention these days. While there are many ways that use EEG data to build the functional network, I will introduce a very powerful method here. Generally speaking, it can be done by taking the following three steps:

- Define the network nodes. (The nodes could be defined as individual electrodes or as anatomically defined regions of interests.)
- Estimate a measure of association between nodes. (This could be the coherence/correlation between two sensors, or the correlation of the region-average signals between different regions. This can be chosen accordingly to different experiment settings.)
- Generate a matrix by arranging all pairwise associations between nodes and (usually) apply a threshold to each element of this matrix to produce a binary matrix.

Besides introducing this method and building a functional network for a behavior task, this project also aims to tell you something about neuroscience. In chapter two, I will give a brief introduction to neuroscience and familiarize you with some of the important concepts and terms in neuroscience. This chapter is based mainly on wikipedia.

In chapter three, we will also wander around in the world of neuroscience. I will talk more about the typical strategies used to build the functional network. You can expect to learn some classical techniques and thoughts that are used in this project and other papers. In this chapter I will also present some of the clinical applications in analyzing the functional network. They are also mentioned in some other papers (R. Matthew Hutchison et al., 2013). These applications can inform you some of the latest achievements in this area.

The sixth chapter is based on the paper by Emily P.Stephen et al. (Emily P. Stephen et al., 2014), it is like a simulation lab. As it is almost impossible to validate this method physically, simulating the data with a known network structure and use this method to try to regain the network structure is a very good way to test this inference method. To better understand the capability of this method, I will also change the amplitudes of the components that consist the simulated data. This procedure can change the SNR and correlation

ratio of the EEG data. By applying the inference procedure to these different signals, we can know how powerful this method is in building the functional network.

In chapter seven, I will apply this inference procedure to the EEG data. Besides doing what is described in the paper (Emily P. Stephen et al., 2014), I will also extend this method to analyze and interpret the functional network given different stimulus. After that, I will apply this method to other datasets and try to find the general pattern of this functional network.

I will use chapter eight to conclude this project with a discussion. I will talk about some of the limitations and potential problems about this project in this chapter. At the end of this project, there are two appendices. Appendix A is used to present the inferred networks for other datasets obtained from other participants, they can give us more information about the real functional network and help us better understand this method. As this project involves a lot of programming and some of the ideas behind the tricks used are hard to understand. So I make the code for this project available online to make it easier for others to understand this project and repeat the inference procedure. Appendix B serves as a guide to get the code.

Chapter 2

Background

This project aims to build the functional network for our brain, which is a very important topic in neuroscience. However, the concept of functional network sounds abstract and is new to many of us. Besides, the experimental procedure, many of the terms used seems also unfamiliar to us. So it would be very helpful to take a journey to neuroscience first to learn more about some of the notions that are relevant to this project. It can give us some insights into this project and help us understand the importance in doing it. This chapter is based mainly on wikipedia.

2.1 Neuroscience

Neuroscience is the scientific study of the nervous system. It is an important branch of biology that involves anatomy, biochemistry, molecular biology, and physiology of neurons and neural circuits. With the emergence of new methods and techniques, it develops very fast over the years. The scope of neuroscience has broadened over time to include different approaches used to study the molecular, cellular, developmental, structural, functional, evolutionary, computational, psychosocial and medical aspects of the nervous system. Ultimately, neuroscientists would like to understand every aspect of the nervous system, including how it works, how it develops, how it malfunctions and how it can be altered or repaired.

The problem this project tries to solve is an important problem in cognitive neuroscience. Cognitive neuroscience focus on analyzing how mental processes are produced by brain activities. The development of cognitive neuroscience is a result of the application of advanced measurement techniques such as neuroimaging (e.g., fMRI, PET, SPECT) and electrophysiology. They provide the possibility for scientists to probe into our brain and analyze abstract questions such as how human cognition and emotion are related to specific brain activities. The findings of cognitive neuroscience can help us better understand our brain, help diagnose and even cure some of the mental diseases.

2.2 Biological Neural Network

Biological neural network is an important concept in neuroscience. It is a series of interconnected neurons that translate the signal to perform certain actions. In biological neural network, the neurons interact with their neighbors though a complex structure. This structure usually consists of several axon terminals connected via synapses to dendrites on other neurons. It functions according to a simple rule: when the sum of the input signals into one neuron surpasses a certain threshold, the neuron will be activated. It will then send an action potential (AP) and transmits this electrical signal along the axon. This process

continues so the signals can be transmitted to all parts of our body.

The functional network is a result of biological neural network, the receiver is activated by the stimulus and the electronic signal is then transmitted by the biological neural network to different regions of our brain. The functional connectivity is thus created and result in the functional network.

2.3 Functional Connectivity and Dynamic Functional Connectivity

Functional Connectivity

Functional connectivity is the systematic associations between neural activities in different parts of brain regions. This activity refers to both resting state and task state. Task state functional connectivity usually means neural activity correlations during a certain task (e.g., speaking, running, etc), resting state functional connectivity focuses on connectivity calculated across individual blood-oxygen-level-dependent time points during resting conditions. Analyzing the functional connectivity can be really rewarding. Someday it may be able to help provide more definitive diagnoses for many mental health diseases and deepen our understanding of many mental disorders.

Dynamic Functional Connectivity

In analyzing the functional connectivity, the scientists have found that the structure of the functional connectivity often changes over a short time. This finding is very different from the traditional understanding of functional connectivity and this phenomenon is then referred to as dynamic functional connectivity (DFC). Analyzing DFC can help us understand many different neurological disorders. And the researches have also shown that DFC is a more accurate representation of functional brain networks. DFC is a recent development within the field of functional neuroimaging and is now receiving more and more attention.

2.4 EEG data

Electroencephalography (EEG) is an electrophysiological monitoring method to record electrical activity of our brain. It is typically noninvasive. It measures our brain activity by placing the electrodes along the scalp, covering the regions where the brain activity is of interest. EEG measures voltage changes caused by ionic current within the neurons in the region of interest.

EEG, and the related study of event-related potentials (ERPs) are used wildly in neuroscience and cognitive science. Because it has many advantages. First of all, EEG has very high temporal resolution. It can record on the order of milliseconds, which is favorable to some special experiments. Typically, EEG is recorded at sampling rates between 250 and 2000 Hz in clinical and research settings, and nowadays, the advanced EEG apparatus can even record at sampling rates above 20,000 Hz, this provides the possibility to catch the transient dynamic changes in our brain. Second, it is extremely uninvasive, this is very important in this project, and the reason will be explained later. What's more, EEG can detect covert processing (i.e., processing that does not require a response), this fits very well with the behavior task. So EEG is a very good choice for this experiment.

2.5 Landolt C

A Landolt C, also named as Landolt ring or Landolt broken ring, is a wildly used symbol used in many experiments and vision tests across the world. It got its name from its inventor Edmund Landolt.

The Landolt C consists of a ring with a gap, thus is similar to the letter C. The gap can be set to many different positions on the ring (usually 8 possible positions, evenly spaced from 0 to 315 degrees in intervals of 45 degrees). In the test, the participant is asked to decide on which side the gap is. The size of the C and its gap will decrease as the test goes on until the error rate of the subject reaches a certain limit. The minimum perceivable angle of the gap is taken as measure of the visual acuity.

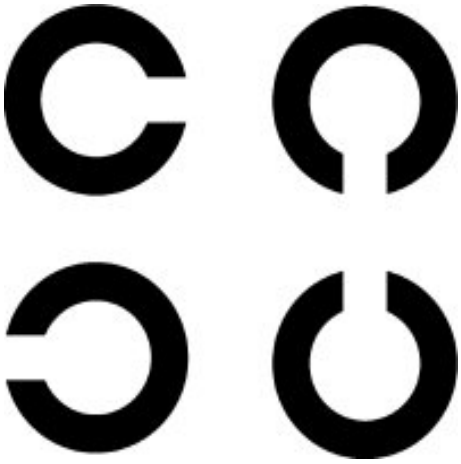

Figure 2.1: Examples of Landolt C

Chapter 3

Typical Strategies and Clinical Applications

As introduced before, analyzing the functional network has been a hot topic in neuroscience for a long time. Many methods to tackle this problem have been proposed over these years and many important applications seem very promising. In this chapter I will introduce some of the effective and wildly used strategies and talk about some of the important clinical applications of analyzing functional networks. This chapter is based mainly on the paper by R. Matthew Hutchison et al. (R. Matthew Hutchison et al., 2013).

3.1 Typical Strategies

Sliding Window Analysis

The sliding window approach has been the most commonly used strategy for inferring the dynamic networks. In this approach, first, a time window of a fixed length will be defined. Then the data points within that window will be used to calculate the functional network. The window is then shifted forward by a fixed number of data points, thus we can infer the functional work at a different time. This process can calculate and present the time-varying behavior of the chosen test statistic. In this project, I also use sliding window analysis to do statistical inference to the dynamic functional network.

The sliding-window analysis is a very good method for exploring dynamic functional network, but we must note that there are some critical issues in using this method. One big problem is that most sources of noise in EEG time series are non-stationary, they are very difficult to eliminate and can cause changes in dynamic functional networks over time. Besides, white noise and other noises that exist in the trial can exhibit fluctuations in common functional connectivity metrics. This can be really bothering in the analysis. As a result, sliding-window analysis should come with hypotheses that are supported with appropriate statistical testing. For example, instead of asking simply whether (and by how much) functional network varies over a scan, one should use statistical test to decide whether the range of sliding-window variability between particular regions is significantly different between two patient populations.

Another thing I want to address about the sliding-window analysis is the choice of window size. The window should be large enough to be able to generate robust estimation of functional network and it should also be small enough to catch potentially interesting transients (Sakoglu et al., 2010). Empirically, window sizes around 30-60s have been noted to produce robust results in conventional acquisitions;

Time-frequency Analysis

Time-frequency analysis is an important technique in signal processing that study a signal in both the time and frequency domains simultaneously. Rather than viewing a 1-dimensional signal and some transform, time-frequency analysis studies a two-dimensional signal – a function whose domain is the two-dimensional real plane, obtained from the signal via a time-frequency transform. Time-frequency analysis is very good for the case when the signal frequency is varying with time. Since many signals of interest–such as speech, music, images, and medical signals–have changing frequency characteristics, time-frequency analysis has broad scope of applications.

As introduced before, a key limitation of sliding-window analysis is the use of a fixed sliding window size. The window size can control the time step of this experiment; Once it is decided, many of the characteristics of this dynamic network is fixed. Yet, both the neurally relevant frequencies and the appropriate time scale for studying connectivity can change during the experiment. A time-frequency analysis can be used to estimate the coherence and phase lag between two time series as a function of both time and frequency. This can free us from selecting a fixed sliding-window size. However, in this project, I didn't use the time-frequency analysis because the time scale I analyze in the project is very short (500ms before the stimulus onset and 70ms after the stimulus onset), there is not so much change in frequency with respect to time.

Independent Component Analysis

In signal processing, Independent component analysis (ICA) is a computational method aiming at separating a multivariate signal into additive subcomponents. To achieve this, this method assumes that the subcomponents are non-Gaussian and are statistically independent from each other. ICA is a special case of blind source separation.

Since the late 1990s, spatial ICA (sICA) has been applied to EEG data to build the functional network. However, in this project, I didn't use independent component analysis, as this method relies a lot of assumptions, while not all of them applies to this project. (e.g., ICA assumes that the values in each source signal have non-Gaussian distributions, however, the white noise I add in the simulated data is Gaussian distributed). Besides, the ICA procedure requires a lot of computation, and the method without using ICA can also give a good result.

3.2 Clinical Applications

An important achievement of studying the functional networks is that we have found the link between some of the physiological and psychiatric diseases and the dynamic functional network. We are not sure if the differences in the functional network is the cause of these diseases, however, this difference is really worth investigation and this research my lead to a better understanding of these diseases and may help us find a cure for them. Here I will present some of the important findings and I hope that more progresses will be made in this field.

Schizophrenia

Many researches have been done to analyze the dynamic functional network of patients

with schizophrenia (Damaraju et al., 2012; Sakoglu et al., 2010). Sakoglu et al. have found that for the patients with schizophrenia, their sensory, motor, and frontal networks had less engagement with other networks (including the default-mode network) when an auditory oddball task stimulus was presented. Besides, they have also found significant differences in the time-frequency patterns of connectivity between schizophrenic patients and the healthy people. Damaraju et al. applied the dynamic functional network approach to a large group ($N > 300$) of patients with schizophrenia and healthy people. The result of K-means clustering of dynamic functional network showed that the functional networks for the patients and the healthy people are similar. However, the functional connectivity of healthy people were found to switch more often. This suggests that patients with schizophrenia tend to linger in a state of "weak" and relatively "rigid" connectivity, while healthy people can quickly switch between different functional networks and are therefore probably faster in dealing with different situations.

Major Depression

The scientists have also found the relation between depression and the functional network. A model that is put forward recently claims that the major depressive disorder (MDD) as an imbalanced state shift that is biased towards being "stuck in a rut" (Holtzheimer and Mayberg, 2011). This model suggests that the changes in functional network could result from a bias towards more depressed states. And it would also be difficult for the patient with MDD to change his mood Initiatively. There are also other findings suggesting that some of the key brain structures in MDD may be altered. This change waken the subjects' ability to react to external or internal cognitive demands, thus making it more difficult to get out of depression (Hamilton et al., 2011; Horn et al., 2010). More work is needed to understand the connections between fluctuating emotion states and brain network dynamics.

Alzheimer's Disease

Many researches have found different functional networks in Alzheimer's patients compared to healthy people (Buckner et al., 2009; reviewed in Greicius, 2008; Menon, 2011). Jones et al. have found differences in the "dwell time" within different sub-network configurations of the default-mode network between Alzheimer's patients and age-matched healthy people. More specifically, there was less time spent in brain states with strong posterior default-mode network region contributions and more time in states characterized by dorsal medial prefrontal cortex component contributions. From this we can know that considering temporal features of functional network may provide a more accurate description of Alzheimer's disease. This discovery may lead not only to a better understanding of this disease, but also to better diagnostic and even potential cure to it.

Chapter 4

Data

In this chapter, I will talk about the dataset used to build the functional neural network. This includes the introduction to the data source, a description of the experimental procedures and the definition of the region of interests. Moreover, because this experiment is conducted in different settings and these settings may have an implicit effect on the functional network, so I will also visualize the data to present some of its marginal distributions.

4.1 Data Source and Experimental Procedures

The data used in this project is obtained from a behavior task in an experiment that is conducted by Jeppe Hoy Christensen from the University of Copenhagen, Department of Psychology. The purpose of this experiment is to analyze our brain activities of visual perception and decision-making processes. (i.e., analyze the data points before the stimulus onset and predict whether the participant will report the right answer in the experiment). Here, I will use this data to build the functional networks before and after the stimulus onset for this behavior task. While this behavior task is mentioned throughout this project, any repeated task with a baseline period could be a good substitution. Preparing for and processing the stimulus involves a large network of brain regions that spans several lobes of the visual system. The functional network can be really complicated and is worth investigation.

During the task the participants was asked to fixate on a centrally located fixation cross on a computer screen. They were instructed to keep fixation during the course of the trial. Then two rings were presented, one on each side of the fixation cross. One ring has a gap (i.e. it's a Landolt C) and the masks are presented for 500ms and a blank screen is shown. Then, the participant has to report back the orientation of the Landolt C by using the numpad keys. (i.e. "1" is south-west, "9" is north-east). They were instructed to only give a response when fairly certain, otherwise they should report that they didn't notice anything. The process of this experiment is roughly shown in figure 4.1. The visual stimulus of this experiment is the gap on the ring and the baseline periods is the time interval -1.5s to -1s relative to the visual stimulus onset.

The experimental procedure is described above, we should note that the experiment is conducted under different conditions.

- Initial fixation was shown for a random duration between 1.5 - 2.5s (uniformly distributed) to introduce jitter for ERP analysis. This means that in the original dataset, the length for each trial is different and the stimulus onsets are at different time points for different trials. However, this dataset is preprocessed before the analysis. Each trial was adjusted to 2.5s long and consists of time points between 1.5s before the

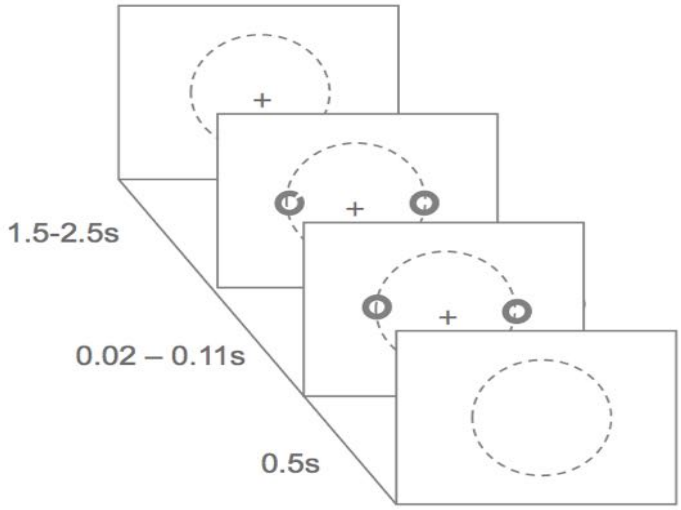

Figure 4.1: The four plots in this figure, from the top left corner to the bottom right corner, represent the process of this experiment. First of all the participant was asked to fixate on the centrally located cross (the top left one). Then after 1.5-2.5s, two rings will show on both sides of the cross and one of the rings will have a gap pointing the eight possible directions (the second plot). Then after a duration of 20-110ms (20ms, 40ms, 70ms, 110ms), the gap will be masked (the third plot). Then after 500ms, the masks will disappear and a blank screen is shown (the last plot)

stimulus onset and 1s after the stimulus onset. The baseline period is between -1.5s to -1s relative to the stimulus onset.

- For each trial, the gap appeared in either the left or the right ring with 50% probability.
- The orientation of the Landolt C was one of 8 possible orientations, evenly spaced from 0 to 315 degrees in intervals of 45 degrees. For each trial, the orientation was randomly chosen.
- The Landolt C was presented for a duration that was randomly chosen among 20ms, 40ms, 70ms or 110ms. Note that in order to have more time points to gain stronger statistical power, in this analysis, I only take trials with a duration of 70ms or 110ms.

4.2 Region of Interests Definition

Analyzing the networks between the different brain regions of interest (ROI) can help us understand our brain networks at the macroscopic scale. Building the network between different brain regions can have three advantages:

1. The connections that are strong across regions but are weak between individual nodes can be detected.
2. Compared to sensor-level functional network, region-level functional network is more robust.
3. Region-level functional networks built by using the same ROIs can facilitate across-subject comparisons.

And to build the functional network between regions, I should define the regions of interests first.

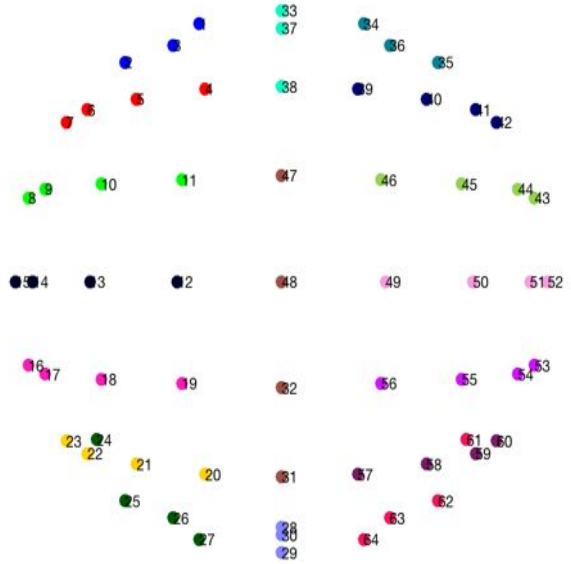

Figure 4.2: The nodes in this figure represents the placement of the electrodes on the scalp. The numbers for each nodes correspond to the rows in the data matrix that contain the data for that node. Different colors represents different region of interests and the nodes in the same region of interests have the same color.

There are many definitions of ROIs in our brain, either anatomically or functionally. Figure 4.2 is the definition of ROIs for this project, different ROIs are indicated by different node colors. We can note that there are 17 ROIs in the figure and each ROI contains three to four nodes. They are distributed all over the scalp and the nodes in the same ROI cover a specific region in our brain.

4.3 Overview of the Data

For this project, there are a total of 16 datasets. Each dataset is obtained independently from a different participant and each dataset contains data recorded from 64 electrodes. The datasets have different number of trials, however, each trial consists of data from -1.5 to +1.0 second relative to stimulus onset and the trials have stimulus durations of 20ms, 40ms, 70ms and 110ms. In this project, I will show the most detailed analysis for dataset 14, the reason will be explained later.

Figure 4.3 is a plot of the EEG data for 10 epochs. All the 64 channels are included in this figure. The stimulus onset and mask onset is indicated by the red and cyan vertical lines in the figure. In this dataset, I have already rejected the epochs that are abnormal due to artifacts (i.e. blinks), but we can still note that in some of the epochs, the data has some abnormal fluctuations. This is very likely to result from some disturbance that is irrelevant to this experiment. They may have a very bad effect on the inference result, so I should try to ”clean” the data before I start analyze it (e.g. standardize the data for each epoch).

Figure 4.4 shows the data points from one trial. We can note that there are 640 data points in this trial, each trial lasts 2.5s and the sampling frequency is 256 Hz. There are many peaks with different interval lengths. They may result from the combination of signals with different periods.

As described above, the experiment settings for this experiment are different for different trials. First of all, I want to specify two things: (1) The orientations of the gap are

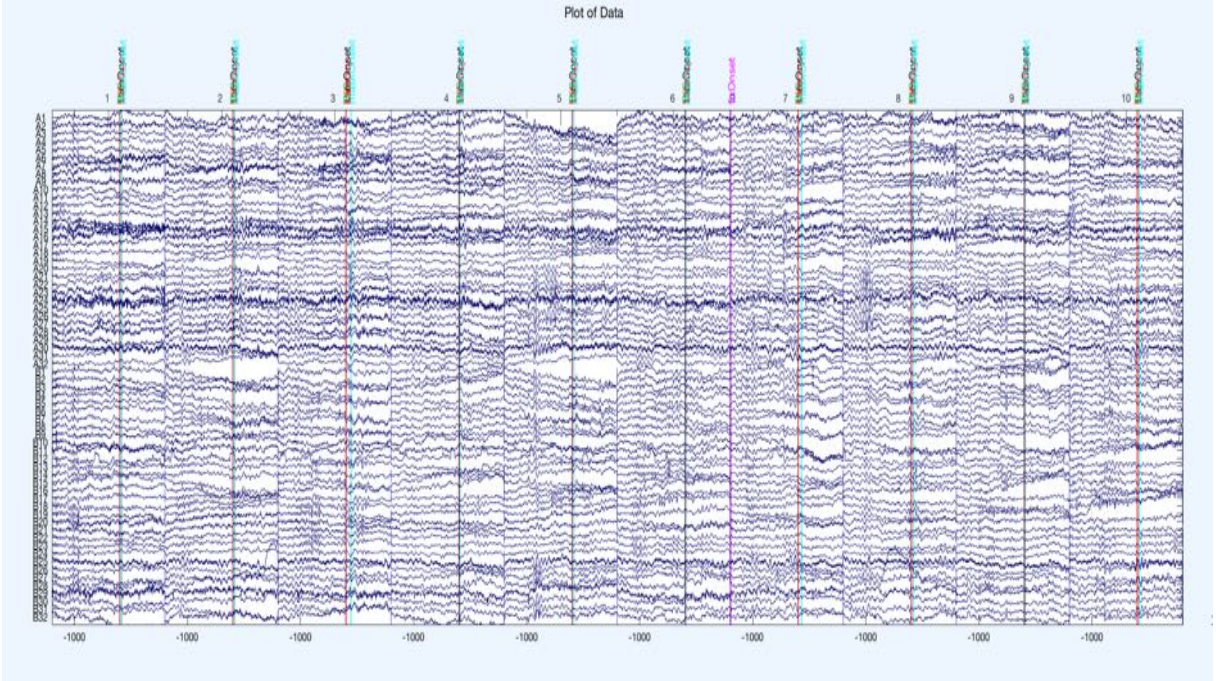

Figure 4.3: This figure is a plot of the data. All the 64 channels and 10 epochs are included in this figure. The stimulus onset and mask onset is indicated by the red and cyan vertical lines in the figure.

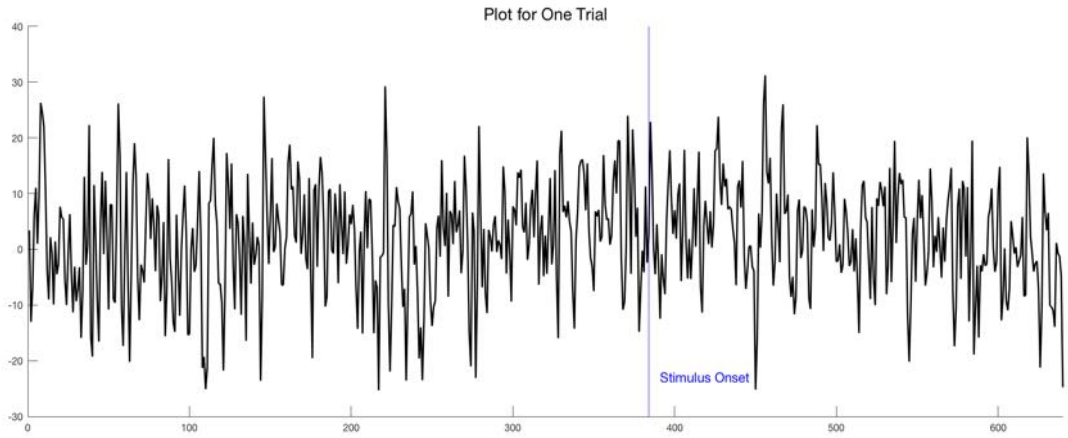

Figure 4.4: The plot of data from one trial, the blue vertical line represents the stimulus onset.

represented by the numbers 1,2,3...8 corresponding to the eight possible orientations of the Landolt rings, i.e., 0 to 315 degrees in intervals of 45 degrees. (2) The accuracies of the trials are represented by numbers 0, 1, 2, 3, 4 and 9. Accuracy level 0 means that the participant gave the correct answer. Accuracy level 1 means that the answer given by the participant is 45 degrees different from the correct answer. 2 means that the reported answer is 90 degrees different from the correct answer and 3,4 means that the mistake is 135°and 180 °. An accuracy of 9 means that the participant didn't give an answer to it.

Here the marginal distributions of these condition variables are presented in figure 4.5. We can note that the positions and orientations of the stimulus are quite uniformly distributed, as discussed in the previous section. From the accuracy figure we can note that in 43.4% of the trials the participant gave the right answer and in 26.7% of the trials the participant didn't give any answer to it. The last graph tells us that in around 250 trials the stimulus duration is 20ms, while other stimulus durations have around 125 trials. However,

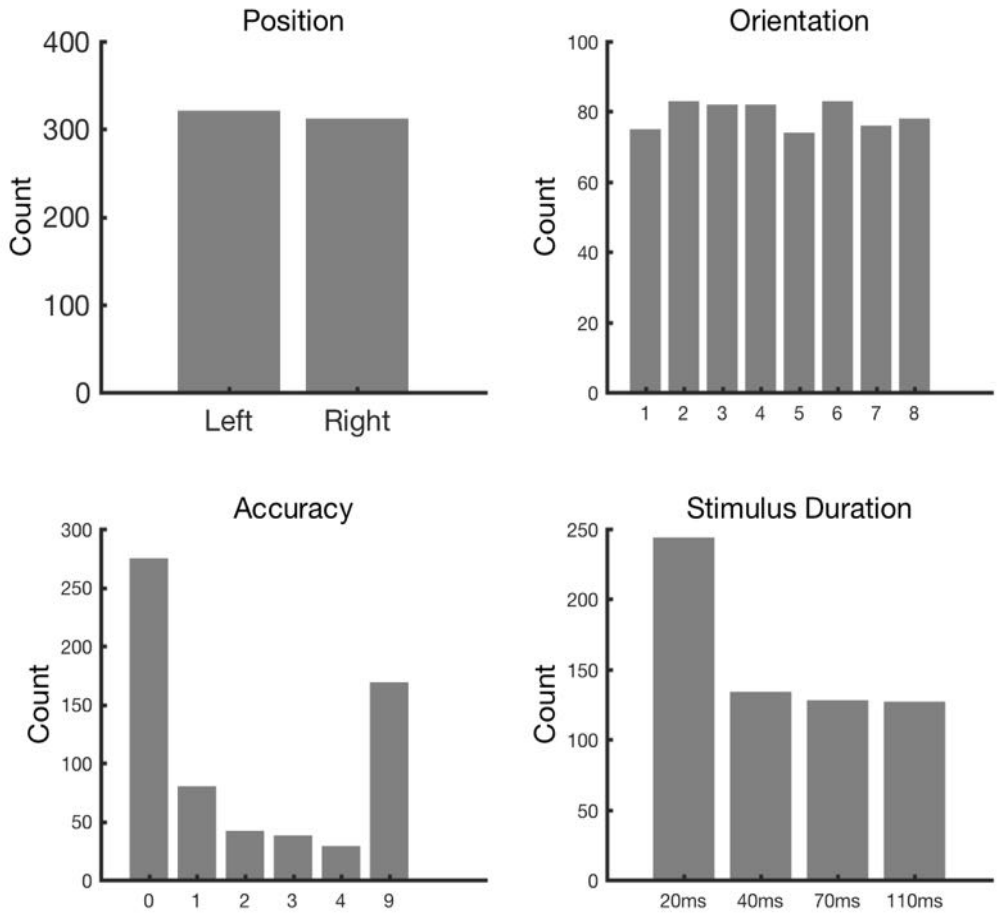

Figure 4.5: This figure presents the marginal distributions of the data. The four plots corresponds to the marginal distribution for different positions of the stimulus, the orientations of the gap, the accuracies of the response and the durations of the stimulus

in this analysis, I only take into account trials with stimulus durations of 70ms or 110ms, so there are 255 trials in the selected dataset.

Chapter 5

Method

The goal of this project is to build the functional network for the behavior task described above, however, it is no easy task. Analyzing and finding the functional connectivity can be really difficult. Here I propose a multi-step analysis strategy, this strategy is based on the paper by Emily P. Stephen et al. (Emily P. Stephen et al., 2014). It tries to find the differences in correlation between the trial and baseline via pairwise statistical tests. The pairs of correlations that show significant difference in the test will be inferred as edges.

To be specific, the inference procedure can be concluded as these five steps:

1. Data preprocessing. First of all, I want to organize the data into the desired forms, making it easier to resample different trials and calculate the test statistics.
2. Computing the test statistic. Then, I calculate the test statistic that measures the correlation between the nodes. This test statistic is very important in determining the edges.
3. Assessing the significance of the test statistic. This step is the statistical test part, it takes the test statistics from the baseline period and trial and assesses whether the test statistics in the trial is significantly different from that of the baseline.
4. Correcting for multiple comparisons. As there are many potential edges, which can lead to many false positives, so here I apply the Benjamini-Hochberg procedure (Benjamini and Hochberg, 1995) to correct for multiple comparisons and control for false discovery rate.
5. Assessing uncertainty. Finally, I will assess the uncertainty of both the individual edge and the network density–the total number of edges in a network divided by the number of possible edges. Assessing the uncertainty can give us a more comprehensive image about the network.

Besides, we should note that network analyses can have a wide variety of forms, here I focus on how correlation-based networks differ during a task compared to a baseline period. We should know that the same framework developed here also applies to other choices of coupling measure (e.g., coherence). The test statistic can be chosen accordingly to achieve the best outcome, however, this won't be discussed in this project. In the following part, I will explain in detail the steps described above.

5.1 Preprocessing of the Data

In order to analyze time-varying functional connectivity, the trial data were then divided into different epochs and analyzed separately. The time points in the first 500ms in each

trial is the baseline data and don't contain any task-related correlated activity. The "before" and "after" epoch (500ms before and 70ms after the stimulus onset) will contain different task-related correlated activities corresponding to the functional network before and after the stimulus onset. To better understand the functional networks and how one dynamically changes to another, I also define a "dynamic epoch", using a sliding window containing 40 time points (10 time points overlap), over trials. By defining this, I can model "dynamic" networks on overlapping time points starting from 120 time points (approximately 500ms) before the stimulus onset until 20 time points (approximately 70ms) after the stimulus onset. The 40 time points "dynamic epochs" were labeled according to their midpoints.

According to the two definitions of epoch, the dataset is grouped in two different ways. For the "before/after" definition, the trial data is separated into before epoch and after epoch with dimension $(T \times N \times L)$ for each epoch. In this definition, T is the number of time points in each trial, N is the number of sensors and L is the number of trials. $((128 \times 64 \times 255)$ for before epoch and $(20 \times 64 \times 255)$ for after epoch). Using dynamic epochs, the data is grouped into 11 matrix of dimension $(40 \times 64 \times 255)$. 11 means there are 11 epochs corresponding to 11 different time indexes for the epochs, 40 time points per epoch, 64 sensors and 255 trials. After arranging the data like this, I then normalize the data by subtracting out the mean and dividing by the standard deviation for each sensor and each trial.

Besides the trial epochs, the corresponding "baseline" intervals were also extracted from the dataset to estimate the null distribution of the test statistic between each pair of sensors. Here, the intervals of baseline data is set to have the same time points as the "before epochs". Then the baseline is arranged in a 3-dimensional matrix with dimensions $(T \times N \times L)$. For example, for before/after epochs definition, the corresponding baseline matrix will have 255 non-overlapping intervals of the (preprocessed) baseline data for 64 sensors, and each interval is 128 time points long. The baseline intervals are then normalized in the same way as the trial intervals: subtracting the mean and dividing by the standard deviation.

5.2 Computing the Test Statistic

Because I am going to find the connections between different nodes/regions, so I need to find a suitable test statistic to measure the correlation between the nodes/regions. In this analysis, two measures of test statistics were used to build the functional networks: correlation (for nodes) and canonical correlation (for regions). The test statistics for all epochs and all pairs of nodes will be calculated and will thus result in a 3-dimensional matrix with dimensions $(E \times N \times N)$ for sensor-level functional networks and $(E \times R \times R)$ for region-level functional networks, where E is the number of epochs and R is the number of regions.

In network, correlations that become more extreme (more positive or more negative) during the task, usually indicates an edge. So here I take the absolute value of the correlation as the test statistic for the sensor-level functional network. This was calculated as:

$$AbsCorr(x, y) = \frac{\mid \sum_{l=1}^{L} \sum_{t=1}^{T} x_l(t) y_l(t) \mid}{\sqrt{(\sum_{l=1}^{L} \sum_{t=1}^{T} x(t)^2)(\sum_{l=1}^{L} \sum_{t=1}^{T} y(t)^2)}}$$

where $x_l(t)$ and $y_l(t)$ are the voltage at time t of the epoch of trial l for sensors x and y, respectively.

In choosing the region-level test statistic, there are two things we must know: (1) For the dataset, which nodes belong to which region of interest is known beforehand. (2) The

data points for all the nodes are available. So I need to make good use of the information I have. Here, I choose canonical correlation as the region-level test statistic. This is because I have already know the grouping of the nodes, it will be very easy to calculate canonical correlation for them. Besides, the canonical correlation can find the maximized correlation for the linear combination of the nodes in a group, this means that this measure take into account the information of all the nodes, the result will be robust and reliable. So canonical correlation is a very good measure for us to turn the sensor-level correlations into region-level correlations. The canonical correlation between two groups of signals $\vec{x} = (x_1, x_2, ..., x_n)^T$ and $\vec{y} = (y_1, y_2, ..., y_m)^T$ is defined as:

$$CanonCorr(\vec{x}, \vec{y}) = max_{\vec{x},\vec{y}} Corr(\vec{a}^T\vec{x}, \vec{b}^T\vec{y})$$

For convenience, I will switch to matrix notation: X and Y are $(n \times LT)$ and $(m \times LT)$ matrices. Each row contains trial voltages for each sensor in the region, m and n are the number of nodes in that region. I can calculate the canonical correlation by using the singular value decompositions of X and Y, $X = U_X \Sigma_X V_X^\dagger$ and $Y = U_Y \Sigma_Y V_Y^\dagger$ where U is an $m \times m$ real or complex unitary matrix, Σ is a $m \times n$ rectangular diagonal matrix with non-negative real numbers on the diagonal, and V is an $n \times n$ real or complex unitary matrix.The canonical correlation is the first singular value of the matrix $Q_{XY} = U_X V_X^\dagger V_Y U_Y^\dagger$.

Another advantage of using canonical correlation is that it can help us find the correlations that are undetectable at the sensor-level network, this can help reduce the false negatives. There are two reasons for the improved performance. First, sometimes the connections between the sensors can be weak and thus can't be detected by the sensor-level functional network. However, if I combine then together using the canonical correlation, the effective SNR and statistical power will be increased, thus make it easier for the true edges to stand out. Second, the region-space network has many fewer edges, it will make the Benjamini-Hochberg procedure to correct for multiple comparisons much less severe. So a higher p-values for the test statistics will be used as a threshold. But on the other hand, even if there is an edge between two nodes, there is not necessarily an edge between the two regions where the two nodes belong. Because there will be an edge if and only if the canonical correlation before/after the stimulus onset between the two regions is significantly stronger than that of the baseline period. This may not be true if only several of the nodes in the two regions are highly correlated in the trial. So sometimes, the sensor-level functional networks and the region-level functional networks can be quite different.

5.3 Assessing the Significance of the Test Statistic

After the test statistics between each pair of nodes are computed, I need to calculate a p-value and run a statistical test to decide the existence of the edges. Here in this project I have chosen a one-sided hypothesis test, trying to find the correlations or canonical correlations that are larger than baseline.

In order to estimate the null distribution, I choose to apply a bootstrap procedure to resample from the baseline data. In this procedure, L baseline intervals are sampled with replacement from the L total baseline trials for each bootstrap sample. Then I calculate the test statistic for each potential edge from the resampled L baseline intervals. Here, I choose to resample 1000 times and calculate the test statistics for these 1000 samples. Thus they build an empirical distribution for the test statistic from the baseline period for each potential edge. This empirical distribution will be used to calculate the p-value and decide whether the edge should exist.

In determining a p-value for each edge, the test statistic is computed for the epoch of interest and compared to the test statistic from the baseline. Specifically, in this project, I define the p-value for the edge between node i and node j as the number of test statistics in the bootstrapped statistics for nodes i and j that is higher than the trial test statistic between nodes i and j, divided by the total number of bootstrap samples (here I use 1000 samples). But we note that when the test statistic is larger than any of the bootstrapped statistics, this p-value will be zero, which is unrealistic. To account for this, I set all zero p-values to be 1/1000, which is the smallest possible p-value for the number of bootstrap samples. Then, I use this p-value to decide whether there will be an edge between the two nodes. If the p-value for an edge is smaller than a certain number (e.g. 0.05) I can decide that this edge exists. Then I apply this method to all pairs of nodes and thus the network can be built.

5.4 Correcting for Multiple Comparisons

In statistics, if we need to consider many statistical inferences at the same time, we need to correct for multiple comparisons to control the false positives. Because this analysis involves statistical tests for many edges, each of which has a potential to produce a "discovery." So multiple comparison is a big issue in this project.

In this project, I decide to use the Benjamini-Hochberg procedure to correct for multiple comparisons and control for False Discovery Rate (FDR). This procedure sets a limit, q (here, 0.05), as the expected proportion of falsely detected edges to the total number of detected edges. The steps of this procedure is described as follows: First, the procedure sorts the p-values in increasing order. Then, it chooses the highest integer k such that:

$$p_{(k)} \leq \frac{qk}{N_{MC}}$$

In this equation, N_{MC} is the total number of tests (which is all the possible edges between all the nodes), and $p_{(k)}$ is the kth smallest p-value. After applying this procedure, only 5% of the inferred edges are expected to be false positives. While on the other hand, a typical p-value threshold of $\alpha = 0.05$ implies that 5% of all possible edges will be expected to be false (which may result in $\frac{N_{MC}}{20}$ false positives, this can be very big when there are many nodes). The Benjamini-Hochberg procedure thus can greatly reduce the number of false positives.

Benjamini-Hochberg procedure is a very effective way to control for FDR. However, it can impose a lower bound on the number of edges that can be detected in the networks, and this lower bound can become really large for a network with many nodes, which can greatly affect this analysis. I will discuss this in detail in the discuss chapter in this project.

5.5 Assessing Uncertainty

The uncertainty of two network features are very important in this project: (1) Edge existence–I want to know more about the existance of every single edge–and (2) network density–an important measure for the functional network as a whole. To assess the uncertainty, I use a nonparametric bootstrap procedure. Specifically, I resample the data from the trials with replacement and then calculate the test statistic for each edge using the resampled data and use then to get the p-value by comparing to the test statistics of the baseline distribution. After this, I apply the Benjamini-Hochberg procedure to create a binary network and also calculate the density of the binary network. Because the data used to build the binary network is resampled from the dataset and this resample procedure can be repeated

for many times and build many different binary network (here I use 100 bootstrap samples). This procedure makes it possible to estimate the probability of each edge. Here I estimate it as the proportion of the inferred networks that contain that edge.This method treats each potential edge as a Bernoulli random variable with parameter p, which is the probability of observing an edge. The variance is therefore equal to $p(1-p)$. By doing so I can determine the variability for an individual edge.

This method can also be used to assess the uncertainty of the network density. For each resampled dataset, the binary network and network density can be calculated, thus I can have an empirical distribution of the network density. Then, I can use this empirical density distribution to estimate the standard error of the density. After that, a 95% confidence interval around the observed network density d_{obs} can be calculated as $[d_{obs}-1.96\hat{s}, d_{obs}+1.96\hat{s}]$.

However, we will note that the estimates of the sampling distributions of networks and network density calculated this way are biased. This is because the resample procedure will lead to fewer unique trials in building the functional network. This will lead to decreased degrees of freedom in the surrogate networks, which may increase the occurrence of false positives. Increasing the number of false positives will tend to increase the probability of individual edges and the density of the surrogate networks relative to the original network. As a result, the estimated probabilities of edges will be higher than the true probabilities and the network densities will also be higher than the true network densities. So in this project I will include the confidence interval for the network density estimation as the confidence interval can better capture the true density.

Chapter 6

Simulated Data

To better understand this inference method and analyze how it behave under different conditions, I will simulate the EEG data with different SNRs and correlation ratios, and then apply the inference procedure described above to them. By comparing the inference results with the true network structure I can test and validate this method. This chapter is based on the paper by Emily P. Stephen et al. (Emily P. Stephen et al., 2014).

6.1 Simulation Procedure

In simulating the EEG data, I expect the correlations between electrodes can be presented as a function of time with respect to the stimulus onset. We know that there should be some task-related activity and connectivity after the stimulus onset, and it's also reasonable to speculate that there are other kinds of connectivity before the stimulus onset, such as processing of the visual stimulus and motor preparation. Hence I consider inferring the functional network before and after the stimulus onset (specifically, 500 ms before until 70 ms after the stimulus onset). In this simulation, I consider the brain activity recorded from eleven sensors. The simulated data at each sensor consists of four components: Pink Noise $P(t)$, White noise $W(t)$, 2-50 Hz correlations that exist throughout the recording, and 8-25 Hz correlations that only exists during the trials. The 2-50 Hz correlations consists of constant correlated structure that is for all nodes $C(t)$ and uncorrelated noise between sensors $B(t)$. The 8-25 Hz correlations consists of task-related correlation structure $T(t)$ and task-related uncorrelated signals $U(t)$. So the signal $X_i(t)$ we observe at node i should be expressed as:

$$X_i(t) = P_i(t) + W_i(t) + C(t) + B_i(t) + T_i(t) + U_i(t)$$

In this chapter I will discuss how each component is generated, added to the specific nodes, time points and how the functional network is detected by this method.

Pink Noise

Pink noise is a signal or process with a frequency spectrum such that the power spectral density is inversely proportional to the frequency of the signal. It is named pink noise because of the pink appearance of visible light with this power spectrum.

Within the scientific literature the term pink noise is sometimes used a little more loosely to refer to any noise with a power spectral density of the form:

$$S(f) \propto \frac{1}{f^{\alpha}}$$

where f is the frequency of the signal, and $0 < \alpha < 2$, with exponent α usually close to 1. Pink noises exists wildly in nature. In this simulation, the "$1/f$" reduction in power captures one feature of brain voltage activity.

The pink noise can be generated as a linear convolution of white noise with a Gaussian kernel. Because it exists though out the experiment and in all electrodes, so it will be generated independently for each node and will be added to all the time points in that node.

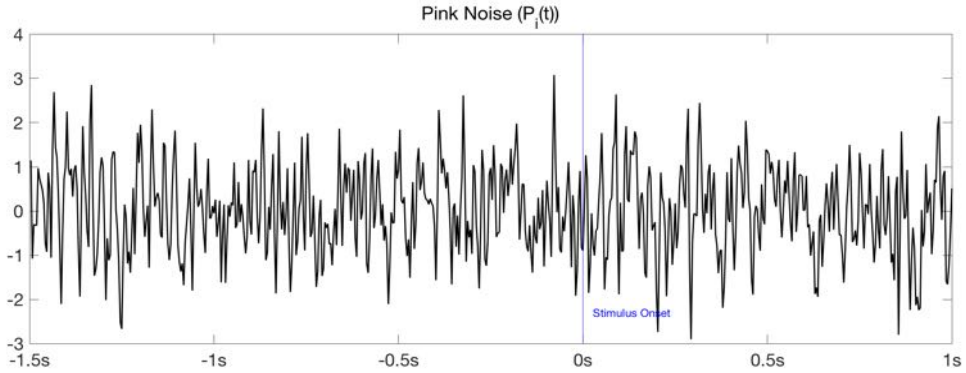

Figure 6.1: Pink Noise $P_i(t)$. In this simulation, N pink noises will be generated independently and added to the data for each node.

White Noise

The second component is white noise, here I use it to represent sensor noise. Like the pink noise, it also exists though out the experiment in all electrodes. So it will be added to all the time points for each node.

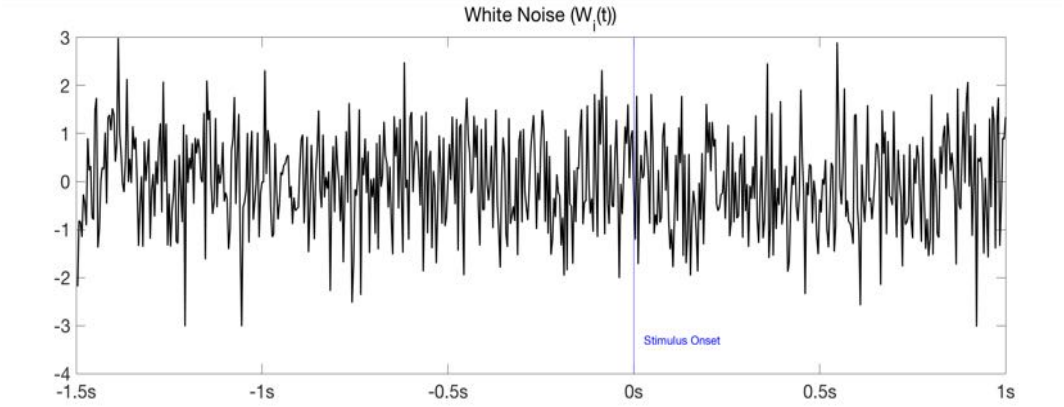

Figure 6.2: White Noise $W_i(t)$. In this simulation, N white noises will be generated independently and added to the data for each node.

2-50Hz Baseline Signals

Besides these two uncorrelated components, 2-50 Hz correlated signals also exist throughout the recording. This is used to represent the persistent correlation existing in the signals (e.g., correlations related to the recording apparatus). This correlated signal consists of two parts. The first part, $C(t)$, is a single 2-50 Hz signal lasting the entire duration of the recording, it is added to every sensor. Because the same signal was added to all sensors, so all the nodes will be correlated. In order to vary the strength of the correlations and at the same

time hold total variance constant, the second 2-50 Hz signal, $B_i(t)$, which is uncorrelated between sensors, was generated independently and added to each sensor. Specifically, if we have eleven nodes, then eleven 2-50 Hz signals ($B_i(t)$ for i = 1, 2, ..., 11) will be generated independently and each of them will be added to one of the sensors. After that, one 2-50 Hz signals $C(t)$ will be generated and added to all of the eleven sensors.

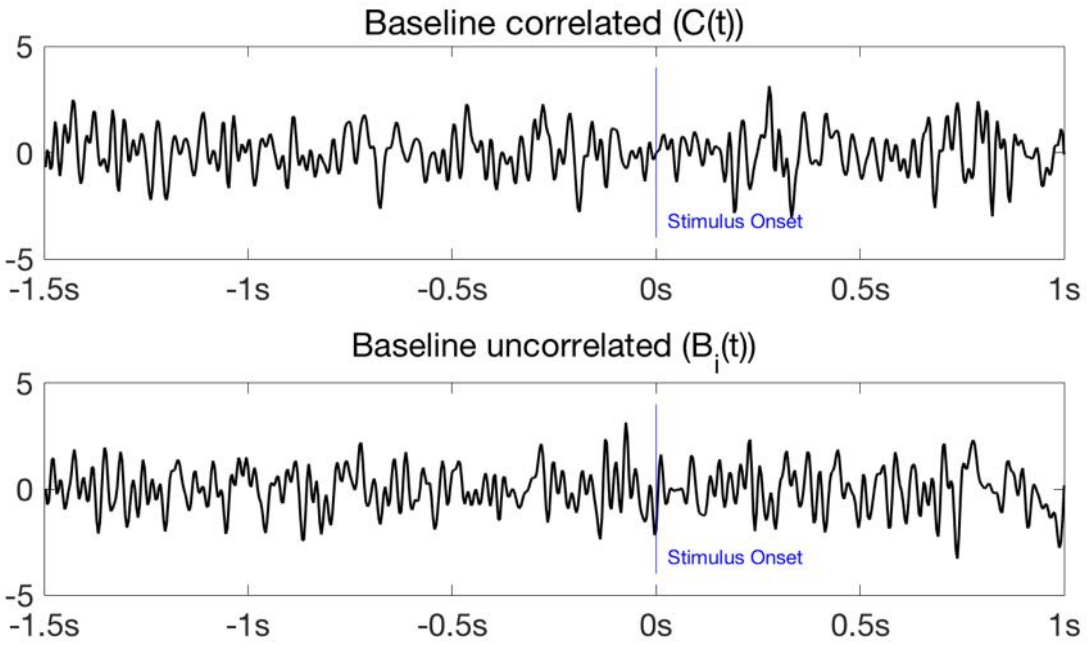

Figure 6.3: Baseline Signals $C(t)$ and $B_i(t)$. N baseline uncorrelated noises will be generated independently and added to the data for each node. One baseline correlated signal $C(t)$ will be generated and added to the data points for all nodes

8-25Hz Task-Related Signals

Besides the signals I mentioned above, 8-25 Hz correlations that only exists during the trials will also be added to the signal. The trial correlations are meant to represent task-related changes in the brain activity. There are two things that are very important about the correlated trial signals. First, they will exist only in specific time points in the trial (500ms before until 70ms after the stimulus onset). Second, the nodes that contain the signals will determine the structure of the network. To better explain this, an example will be used below.

Consider a simple network we have in figure 6.4. To simulate a network like this, we must figure out: (1) which nodes are connected, and (2) is it before the stimulus onset or after the stimulus onset. After finding this out, we should simulate the 8-25 Hz task-related signals and put the same task-related signals in corresponding nodes in the corresponding time points (before/after). In order to hold total variance and spectral characteristics constant, we should put 8-25 Hz uncorrelated signals in the time points where the task-related signals don't exist. When generating the correlated signals that only exist during the trials, we must make sure that the electrodes that are connected in the network will have the same task-related correlated signals. And we also need to note that the task-related correlated signals only appear during a certain time. We should be really careful in generating signals like this.

For example, the task-related signals for the network in figure 6.4 is shown in figure 6.5.

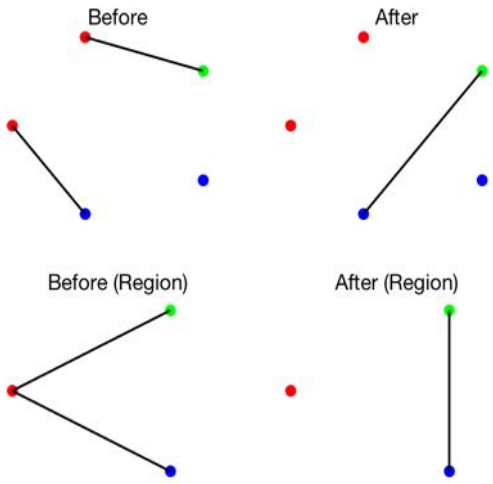

Figure 6.4: This figure presents an example of a functional network. The first row presents the sensor-level functional network and the last row presents the region-level functional network. The regions are color-coded in this figure

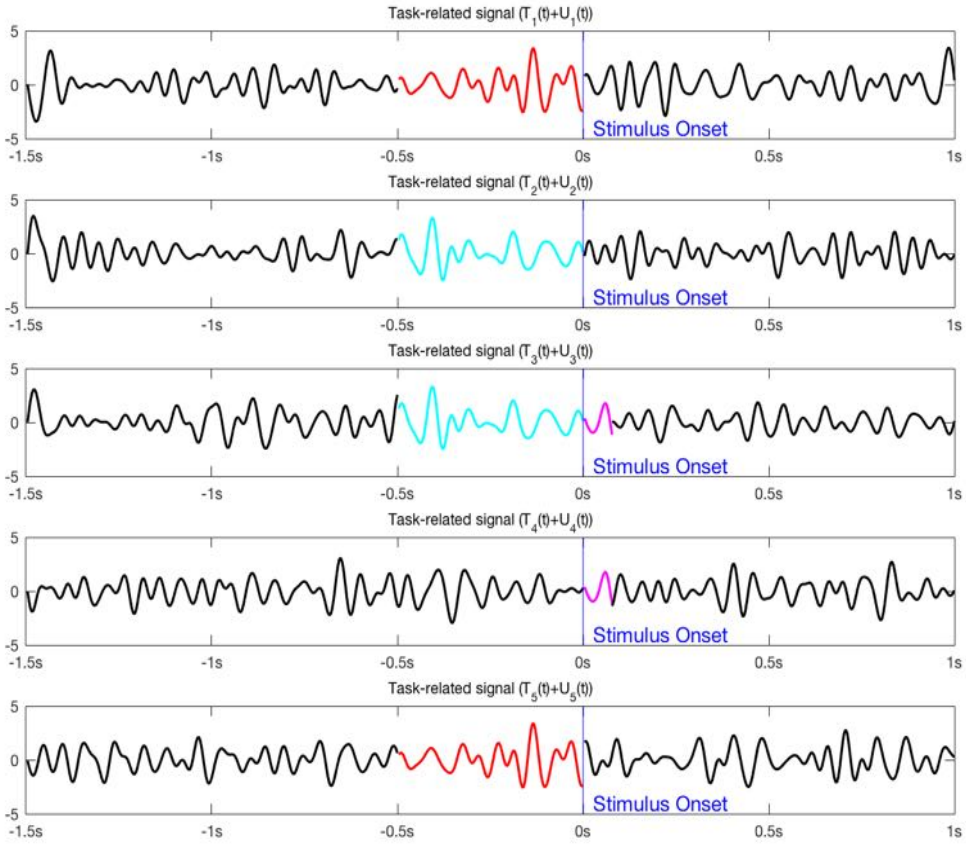

Figure 6.5: Task-related signal $(T_i(t) + U_i(t))$. In this figure, the black curves are uncorrelated signals and the red, purple, cyan curves are task-related correlated signals. The task-related correlated signals with the same color are the same, resulting in a strong correlation between the two nodes with the same signal

In this figure, the black curves represent the uncorrelated signals, the task-related correlated signal (indicated by different colors) is added to the corresponding nodes at specific time points. Thus the signals in these nodes during these time epochs will be correlated and the

edge can be detected by this method.

By summing up these four components together we can get the data we need (figure 6.6). This should be similar to the EEG data obtained from the behavior task. Then I just need to apply the inference procedure to all pairs of nodes and then I can get the inferred network.

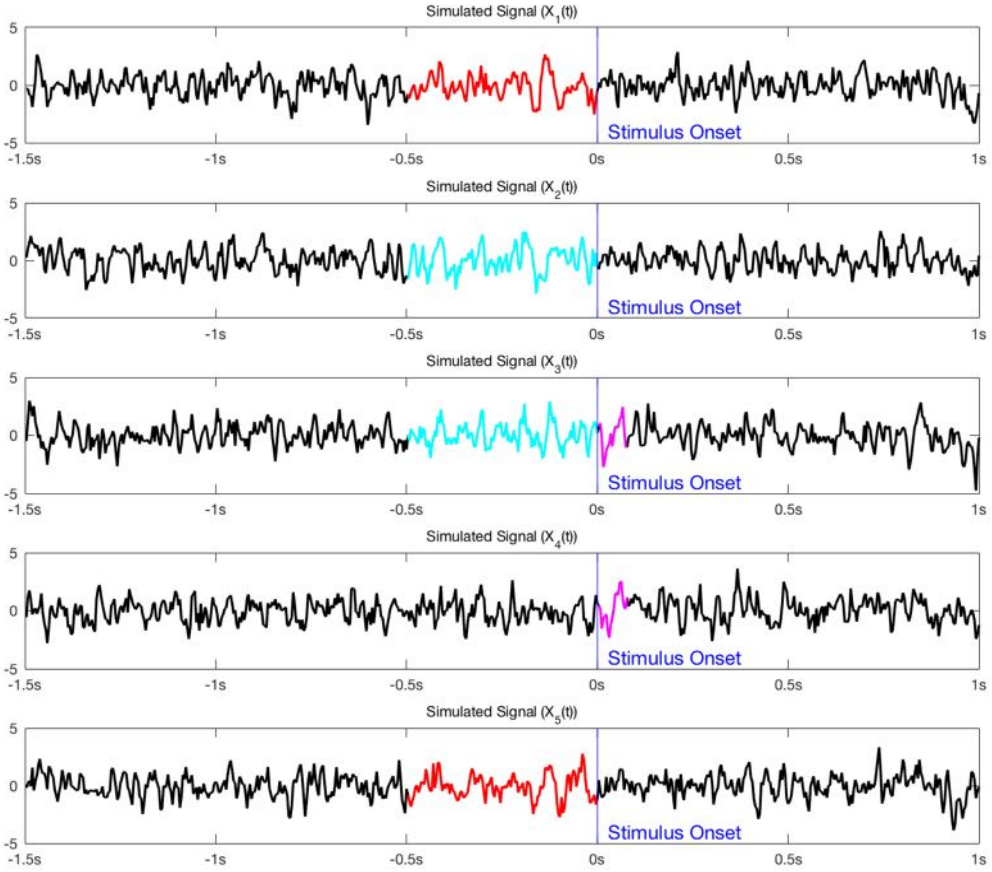

Figure 6.6: Simulated signal($X_i(t)$). This signal is the sum of all the components. The red, purple and cyan curves contain task-related correlations, the goal of this method is to detect the correlation between the curves with the same color.

We should notice that the data is simulated by ourselves and we can control the amplitudes of each component by ourselves. So we can change the amplitude of each component and obtain data of different SNRs and correlation ratios. We can thus test this method to see what the inference results are for datasets with different SNRs.

6.2 Inference Results for the Simulated Data

In this section I'm going to apply the network inference procedure described above to construct functional networks for the simulated data. As mentioned before, I have well control of the simulated data, so I can change the variance of each component to vary the SNR and the correlation ratio. By applying this method to these different data and comparing the results with the real network, I can test and validate this inference procedure.

6.2.1 Definition of SNR and Correlation Ratio

Signal-to-noise ratio is a measure that compares the level of a desired signal to the level of background noise. As introduced before, the synthetic signal consists of pink noise $P(t)$, white noise $W(t)$, constant correlated structure that is for all nodes $C(t)$, uncorrelated noise between sensors $B(t)$, task-related correlation structure $T(t)$ and task-related uncorrelated signals $U(t)$. Among these signals, only the task-related correlation structure $T(t)$ can provide us with the information to build the functional network. Other components can be considered as background noises. So, here, the SNR is defined as:

$$SNR = \frac{Var(T)}{Var(P) + Var(W) + Var(U) + Var(B) + Var(C)}$$

Where the variances are calculated using the data points during the trials (both before the stimulus onset and after the stimulus onset). In simulations to test the inference results for data with different SNRs, I don't want to take the background correlation into account, so here I set $C(t)$ and $B(t)$ to zero. Note that in all the simulations, the white noise component $W(t)$ had a variance of 0.1 and the pink noise component $P(t)$ had a variance of 1. So the only components that can change the SNR are the 8-25 Hz components $T(t)$ and $U(t)$. If I change the variance of $T(t)$ and $U(t)$, then the SNR will be changed.

Besides analyzing this method for different SNRs, I will continue to investigate the effects of constant background correlations that exist throughout the experiment on the inferred trial network. In order to guarantee that the difference isn't result from the change of SNR, for all these simulations I use the same gains for the 8-25 Hz components $T(t)$ and $U(t)$, and the 2-50 Hz component, $C(t) + B(t)$. The only thing that changes are the variances of $C(t)$ and $B(t)$ (still, the variance of $C(t) + B(t)$ will be fixed). As a result, the SNR will be still the same and the ratio of the variance of the trial correlations $T(t)$ to the variance of constant correlations $C(t)$ will change. Here I define the correlation ratio as:

$$CorrelationRatio = \frac{Var(T)}{Var(C)}$$

Here the variances are also calculated over time during the trials. From this formula we can note that a small value of the correlation ratio represents weaker trial correlations relative to background correlations.

6.2.2 Inference of Functional Network Before and After Task Onset

Signals With Different SNRs

As introduced before, for this part, I will consider the scenario in which there is no background noise and the only correlation between the nodes is the task-related correlation. I perform these simulations for four different levels of signal-to-noise ratios (0, 0.05, 0.10, 0.15). Functional networks were constructed using both the correlation and canonical correlation for these simulated data before and after the stimulus onset. As expected, when the SNR is sufficiently large, the network inference procedure can successfully identify the correct functional network.

Figure 6.7 presents the plots of the networks inferred at different SNRs and the true network. The first two columns are the binary functional networks before and after the stimulus onset, the two columns on the right correspond to the probability functional networks before and after the stimulus onset. In this figure, probabilities are indicated in grayscale from 0 (white) to 1 (black), a darker color indicates a higher probability. The five plots in a column correspond to four different levels of SNR: 0.00, 0.05, 0.10, 0.15 and the true

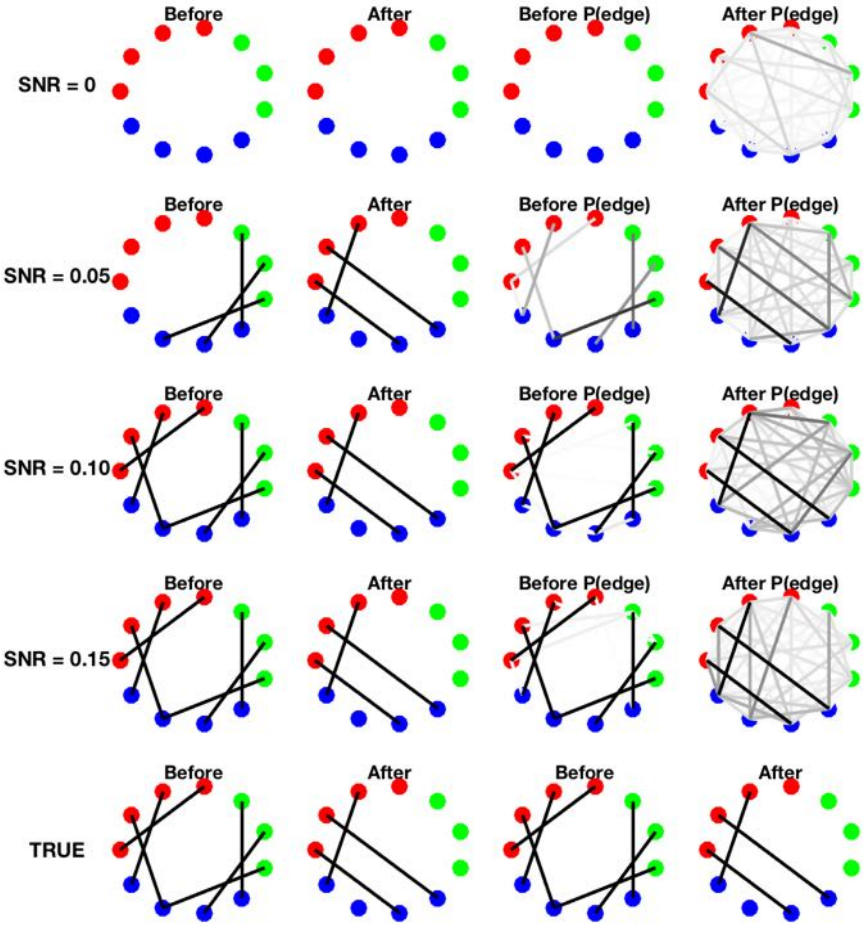

Figure 6.7: Inferred sensor-level functional networks for different SNRs

functional network. In all networks, nodes are color-coded according to region.

We can note from the figure that if the SNR equals to zero, which means there should be no task-related correlated signals in the data, the chance that an edge will be detected is very small. However, we can also notice that the chance for detecting spurious edges after the stimulus onset is much higher than the chance to detect spurious edges before the stimulus onset. This is because there are only 20 time points after the stimulus onset, while there are 128 time points before the stimulus onset. So the results after the stimulus onset will be more unstable and more susceptible to outliers, and it's less likely that spurious edges will show up before the stimulus onset due to its stronger statistical power.

If I rise the SNR, the binary network will have good prediction. When the SNR is higher or equal to 0.10, the binary networks can always identify the true functional network structure. The probability network is also good, as long as the SNR is high enough, all the true edges will show up in the inferred network with a high probability. Spurious edges will also pop up, yet with a comparatively small probability. This result shows that this network inference procedure has a good chance to identify the true functional network when no correlated baseline signals exists between the nodes, even when the SNR is relatively small (0.1).

Then I continue to analyze the region-level functional network for different SNRs. The results are presented in figure 6.8. The result is similar to the sensor-level network. When

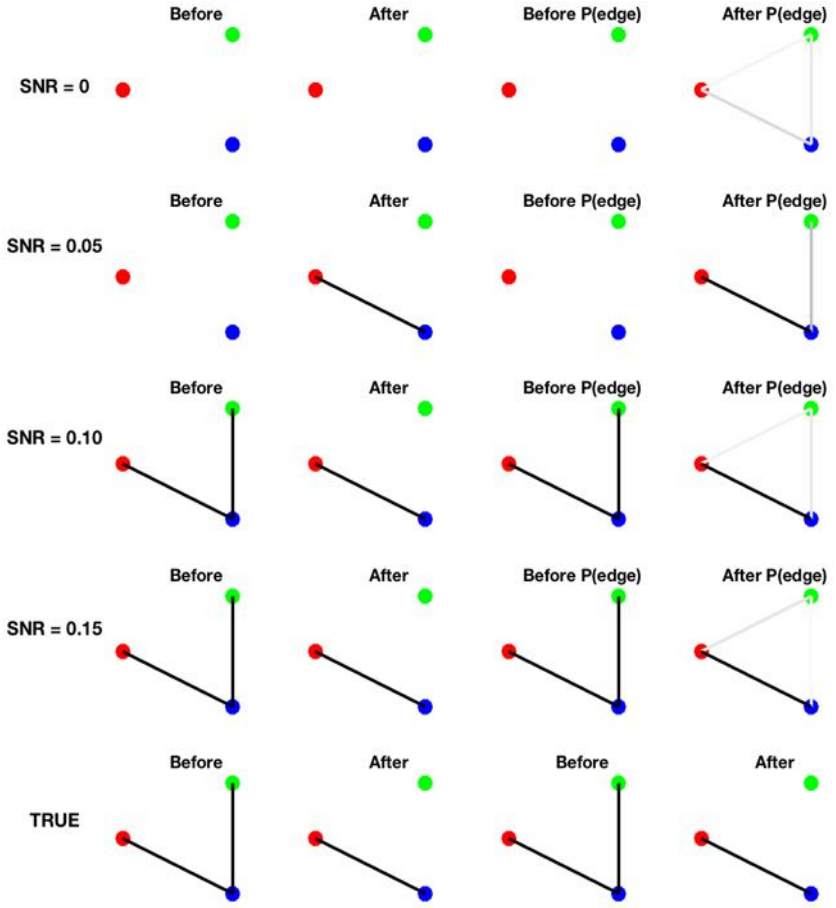

Figure 6.8: Inferred region-level functional networks for different SNRs

the SNR is 0, it is difficult to detect an edge, and when SNR is large, the binary networks can always detect the right edge and the probability network will also have high probability to find the right edges. And also, the functional network after the stimulus onset suffer more from spurious edges due to less data points. Besides, by comparing this region-level network with the sensor-level network, we can notice that the results for the region-level network is more robust. The true edges will always show up with a high possibility and the probability to detect a spurious edge is very small.

To better summarize the changes in functional network from before the stimulus onset to the interval after the stimulus onset, I decide to compute and compare the network density. For the true network, the density is $6/55 = 0.109$ before stimulus onset, and $3/55 = 0.055$ after stimulus onset. At each level of SNR, the density for the functional networks is easily computed from the resulting binary networks in figure 6.7 (SNR = 0.00, before = 0, after = 0; SNR = 0.05, before = 0.055, after = 0.055; SNR = 0.10, before = 0.109, after = 0.055; SNR = 0.15, before = 0.109, after = 0.055). In addition to this single density value, I also compute the confidence interval for each density value (Figure 6.9). In this figure the bar height represents the average of the inferred network densities, the whiskers indicate 95% bootstrapped confidence intervals. The true values of the density before and after the stimulus onset are indicated by the little diamonds. Here, I use the bootstrap procedure described before to compute these confidence intervals. We note that as the SNR increases, the densities before and after the stimulus onset also increase. In addition, the confidence

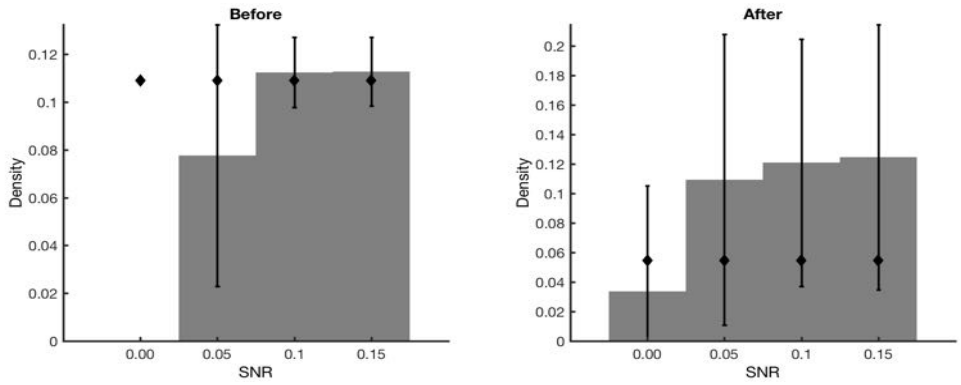

Figure 6.9: This figure presents the network densities for the inferred functional networks at different SNRs. In this figure the bar height represents the average of the inferred network densities, the whiskers indicate 95% bootstrapped confidence intervals. The true values of the density before and after the stimulus onset are indicated by the little diamonds.

interval of trials of SNR equal to 0.05,0.10,0.15 all includes the true density. Besides, at larger SNRs (SNR = 0.10 and SNR = 0.15), the real density and the averaged density are very close and the confidence interval is pretty small, which indicates an accurate and stable estimate. For the after part, things are more interesting. We can note that all the confidence intervals includes the real density, this results from the large scope of the confidence intervals. This is because the inference results are very unstable (we can also note this from the plot of the network in figure 6.7). This is not what I hope to see but it won't make this method useless. As seen in figure 6.7. No spurious edges will show up with a high probability, the average density is high because different spurious edges will pop up in different trials. And on the contrary, the true edges will show up with a high probability, so if I use a threshold to filter the probability network, I can identify the true network structure. The network density plot for the region-level network is similar to figure 6.9 and is not included here.

Signals with Different Correlation Ratios

For the first part I can conclude that this method works pretty well if there is no persistent correlations between the nodes even at a relatively low SNR. Next I will analyze how this method behaves when a constant correlated signal exists in the simulated data. I test it on the signals with correlation ratios 0.5, 1 and 2.

From figure 6.10 we can note that the performance is also good, all the inferred binary networks agree with the true network. However, the probability network is very interesting, it shows that the spurious edges have a higher chance to be detected when the correlation is high (1 or 2) compared to when it is low (0.5). I think it is because the presence of background correlations causes the baseline correlations to be higher, making it more difficult for spurious correlations to appear as false positives in the network estimation. Hence, the presence of background correlations may in some cases make network inferences more robust in this method. Besides, we can also note that even if many edges show up in the probability network after the stimulus onset, the probability of the spurious edges is still much lower than the probability of the true edges and the true edges will always have a high chance of appearance for each correlation ratio. This is different from the different SNR simulations where true network can also have small probability when the SNR is low.

Then I keep analyzing the region-level network for different correlation ratios. We can note in figure 6.11 that the result is very good. The true edges are always detected in the binary networks. The probability networks also testify the result. Spurious edges have a

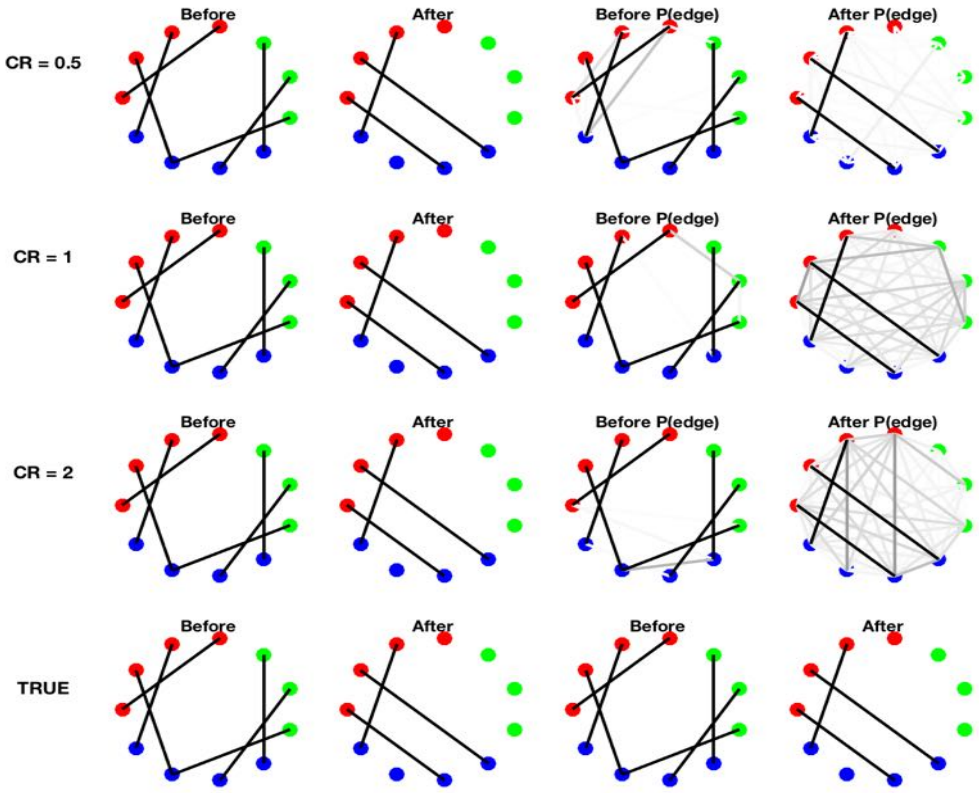

Figure 6.10: Inferred sensor-level functional networks for different correlation ratios

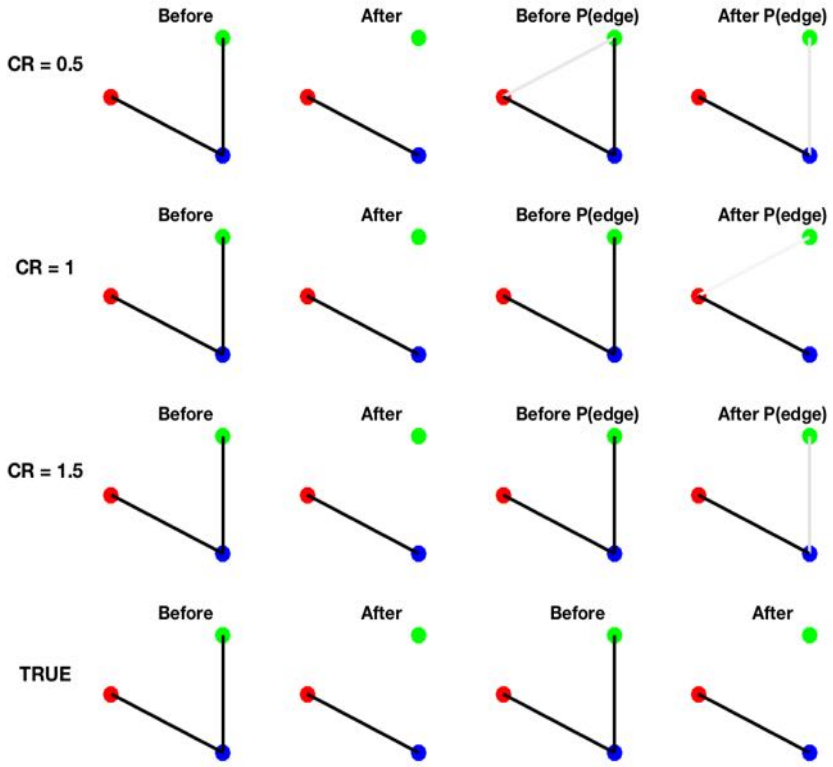

Figure 6.11: Inferred region-level functional networks for different correlation ratios

very low probability to show and true edges are always highly probable. So we can conclude that the canonical correlation can make the region-level networks more robust to spurious edges.

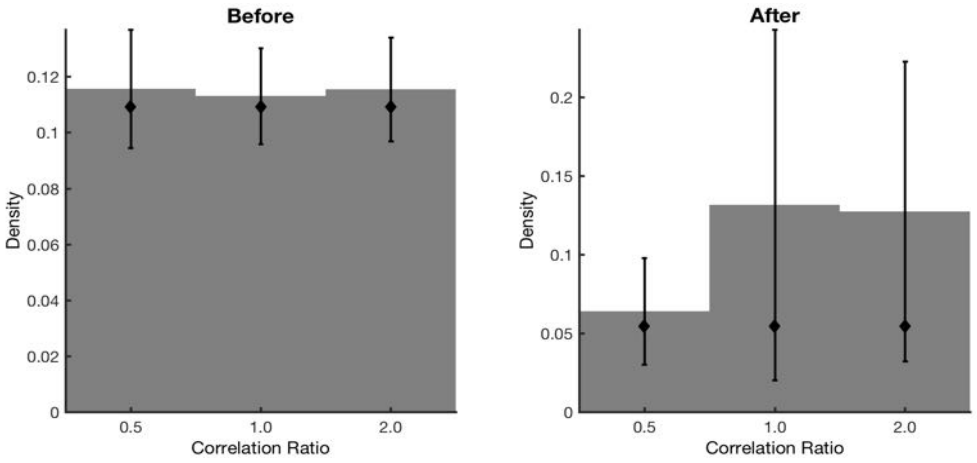

Figure 6.12: Network densities for different correlation ratios.

Analysis of the network density in the presence of constant correlations reveals that the 95% confidence intervals of the density contain the true density in all cases. And again, the density estimate before the stimulus onset is more accurate and stable than after the stimulus onset. There is not much difference in the network density plot before the stimulus onset since few spurious edges show up before the stimulus onset, the inference results are always good and similar for the three correlation ratios. And again, the network density plot after the stimulus onset is very different from the density plot before the stimulus onset. When the correlation ratio equals 0.5, this method will have the best estimate, which is also indicated in the network plot, while many false positives show up when correlation ratio equals 1 and 2, resulting in a high average density estimate and a large confidence interval. But as discussed in the SNR section, if I use a threshold to filter these edges, I can also find the true network for correlation ratio equal to 1 and 2.

We note that these simulation results are from a single instantiation of the simulated signal consisting of multiple trials. But even if I simulate this data for many times and repeat this procedure on them, we can still find similar qualitative results: the network inference procedure can correctly identify the true underlying network. What is different across these instantiations is the number and location of false positive and false negative edges.

6.2.3 Inference of Dynamic Functional Networks

Another important thing I am going to investigate is the dynamic (i.e., time-indexed) functional network. As introduced before, this is solved by using the dynamic epoch (Sliding windows each containing 40 time points). Within each sliding window, I apply the network inference procedure described above. Then I can get the binary functional networks and the network densities with an associated confidence interval.

The dynamic functional network for different SNRs is shown in figure 6.13. In this figure, density of observed networks is presented as a function of time respect to the stimulus onset, with 95% bootstrapped confidence intervals (whiskers). The true density is indicated as the diamonds in the figures. The time indexes for the four example binary networks are -90, -60,

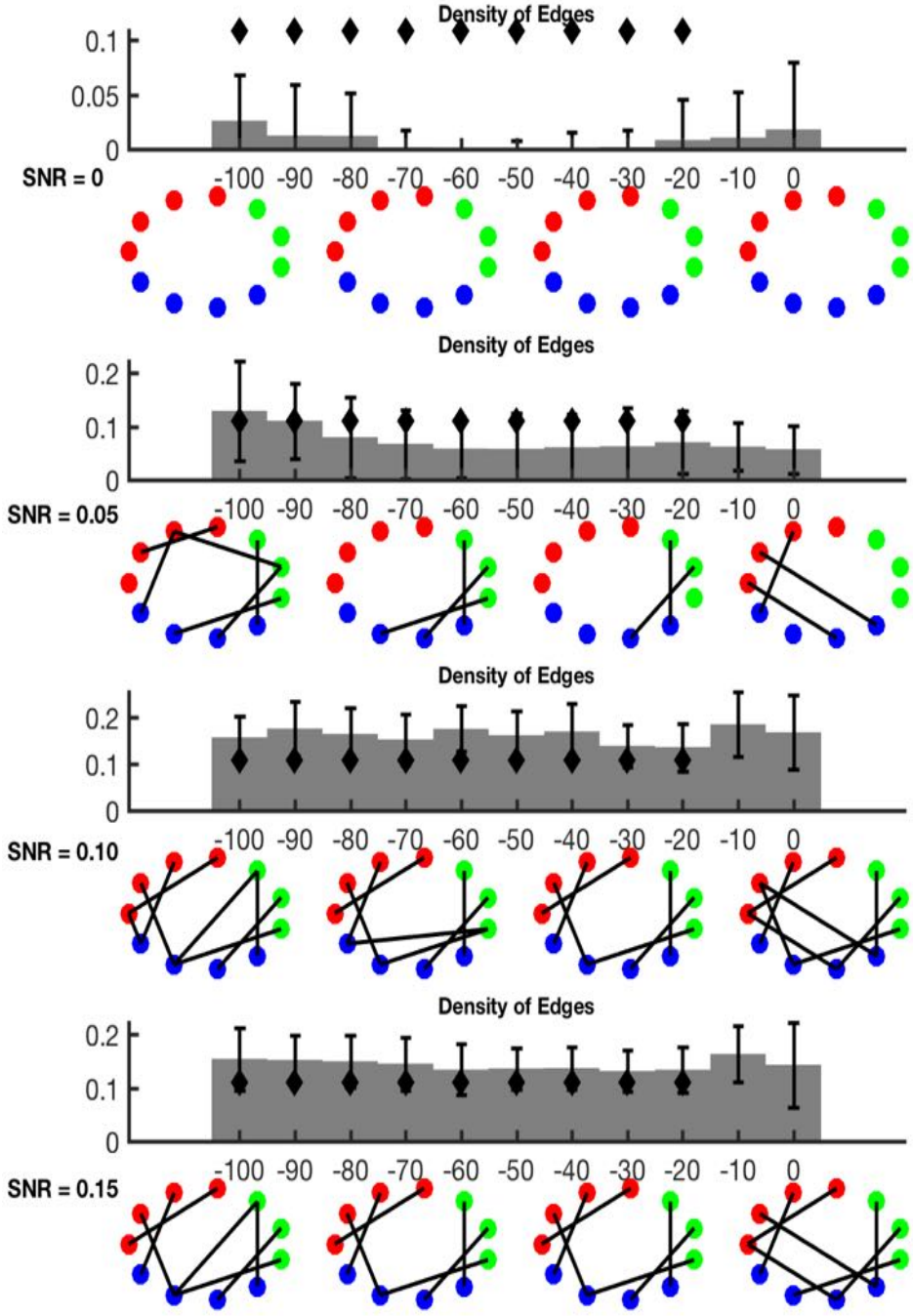

Figure 6.13: This figure presents the dynamic functional network for different SNRs. The x-axis is the time index relative to the stimulus onset. The four binary networks have time indexes of -90, -60, -30 and 0

-30 and 0. This figure shows us how the network density and the binary network change with time. We can note from the figure that the true network density at -10 and 0 is not shown (there are no diamonds there). This is because the data used to infer the functional network at time index -10 and 0 includes data from both before the stimulus onset and after the stimulus onset, so it's very difficult to determine the "true" network density for this situation. And to my surprise, the inferred network densities at these time points seems to be the highest in this process. This could be explained by the plot of the binary network at time point 0. We can note that the network at time 0 includes the edges that appear before

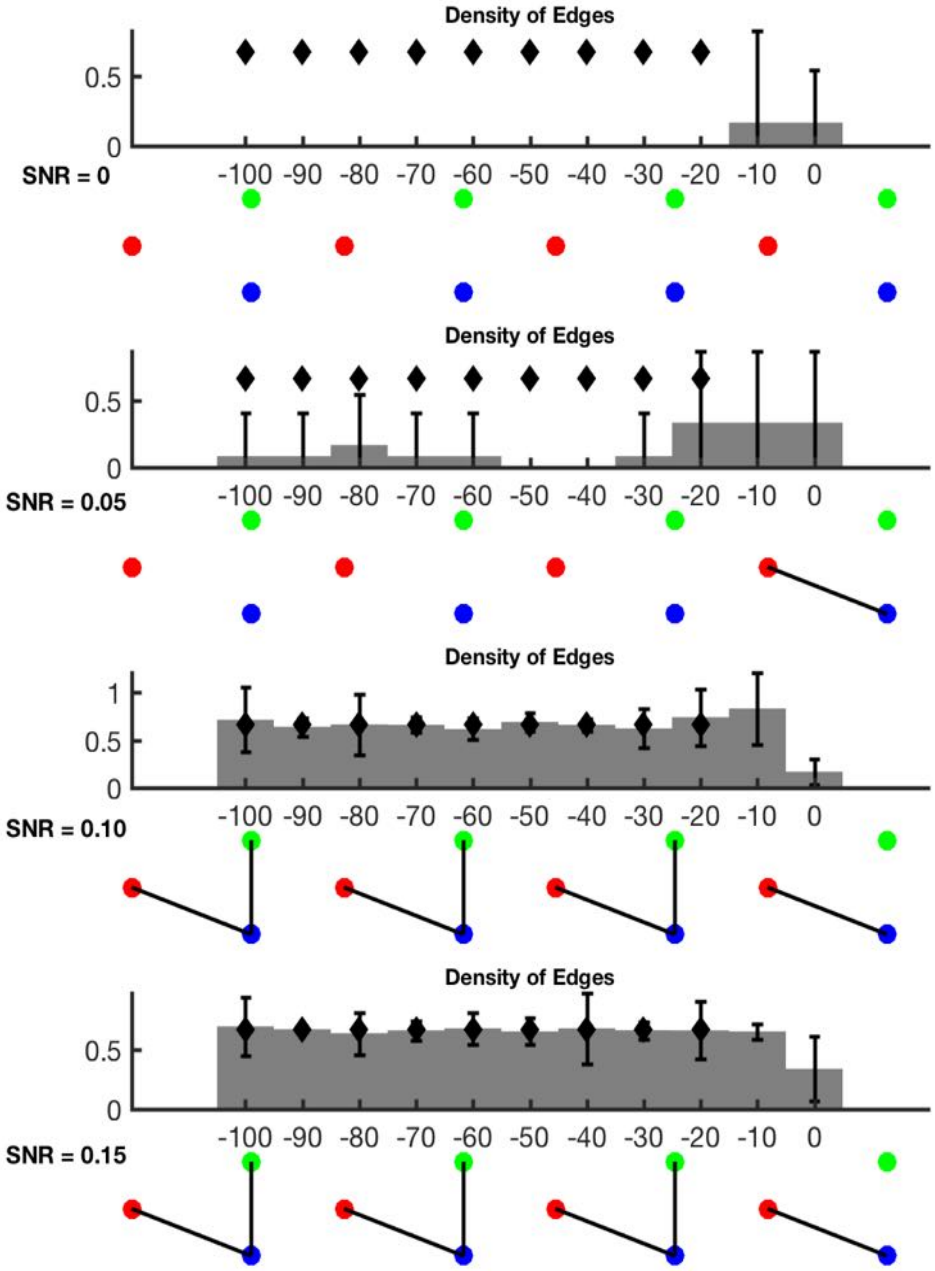

Figure 6.14: The Dynamic Functional Network for Different SNRs (Region)

the stimulus onset and the edges that appear after the stimulus onset, thus it tends to have a high network density at that point.

From this figure we can also note that only several of the edges are detected when the SNR is small. As the SNR increases, the inferred functional networks more accurately capture the true edges, however, more false positives also show up, which lead to an estimated density that is higher than the true density. Besides, the confidence intervals in this figure are very big, this may because I only have 40 time points in each epoch, the inference results won't be as stable if there are many time points.

The inference results for the region-level dynamic networks with different SNRs are in

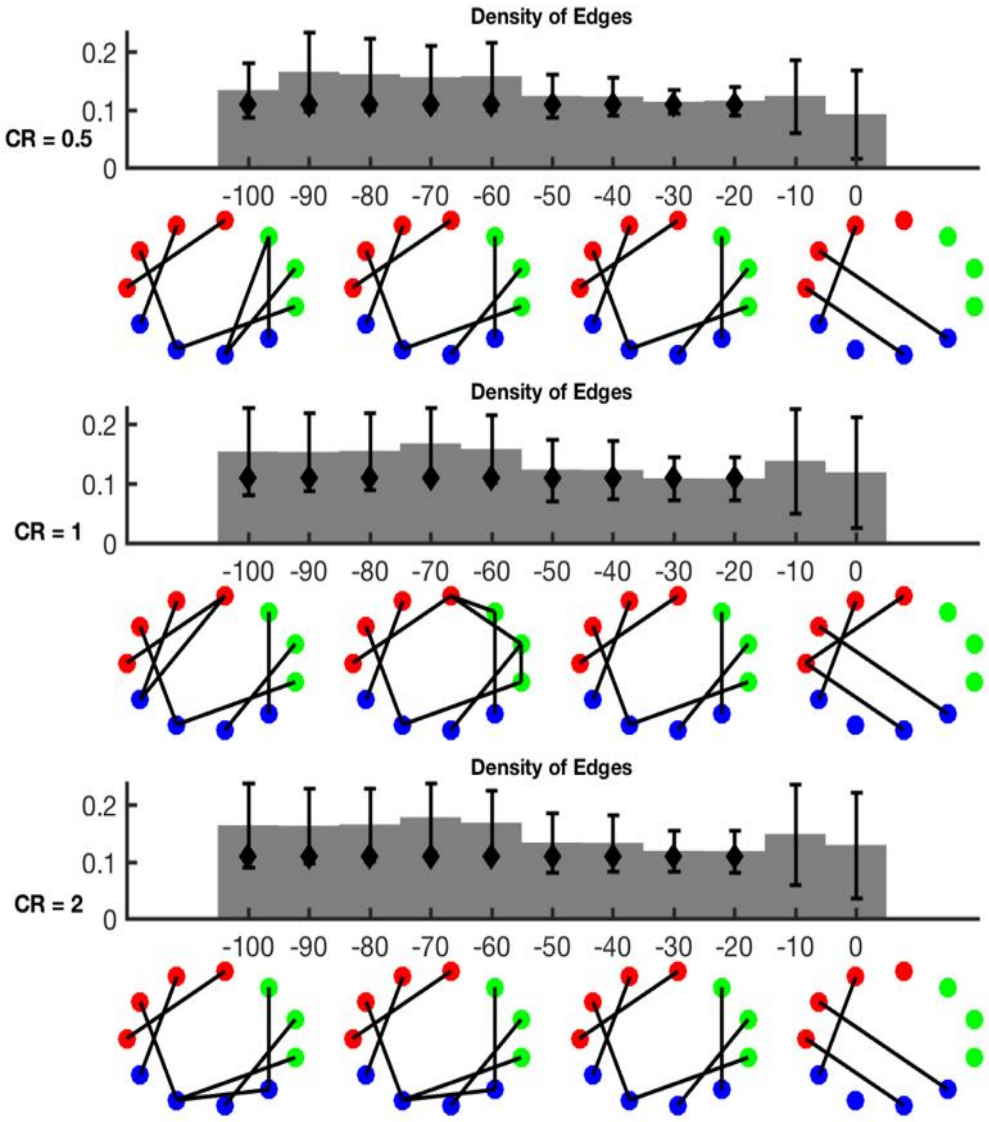

Figure 6.15: The Dynamic Functional Network for Different Correlation Ratios

figure 6.14. We can note that the region-level network is more stable and captures the true density better than the sensor-level network. And we can also notice that the inferred binary networks for SNR large enough (0.1 and 0.15) are the same as the true network, which is better than the inference result of the sensor-level network. This further verifies the robustness of the functional network inference method for regions.

When constant correlations are introduced, the results are also good, especially the network with correlation ratio equal to 0.5. The average densities captures the true density very well and the confidence intervals are small. The results for correlation ratio equal to 1 and 2 is a little worse, the average densities fluctuate more and the confidence intervals are larger, but the true density is always included in all the confidence intervals and the binary networks inferred also suggest good results. Besides, we can notice a upward bias in the density estimation for all correlation ratios, the reason for this phenomenon is explained before.

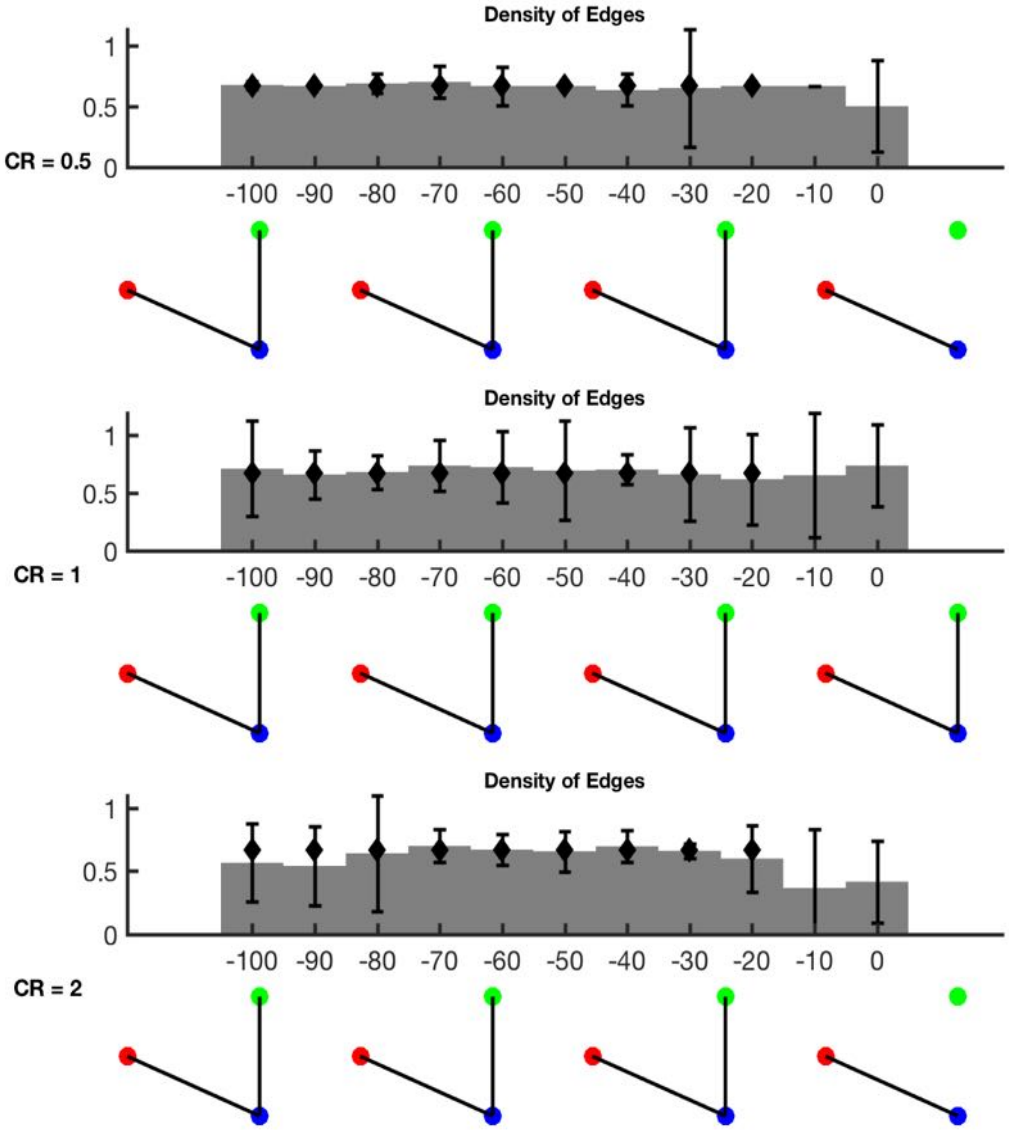

Figure 6.16: The Dynamic Functional Network for Different Correlation Ratios(Region)

The inference results for the region-level dynamic network with different correlation ratios are in figure 6.16. From this figure we can note that almost all the binary networks regain the true network structure. For correlation ratio equal to 0.5, some of the confidence intervals for network density even have a length of 0. This result suggests good performance in inferring region-level dynamic functional network with background correlations.

Chapter 7

Analysis of EEG Data

In this chapter I'm going to apply this inference method to the real data obtained from the behavior task described in the data chapter. I will try to find the real functional networks before and after the stimulus onset. This is a challenging task as there are 64 nodes and 2016 potential edges in the network, which can result in a very complex network structure. Besides, the data is not as "good" as simulated, there can be many artifacts and unexpected disturbances that make the real network hard to detect. However, given the results I have shown above, we can still have confidence in this method and try to regain the real functional network. As introduced before, there are 16 datasets for this analysis, and here I choose to start with dataset 14. I will first focus on the visual cortex, trying to build the functional network among the electrodes within the visual cortex. After that I will continue to analyze how different positions of the stimulus result in different functional networks and how different functional networks lead to different experimental results. Then, I will build the sensor-level and region-level functional networks for the whole brain. At last, I will apply this inference procedure to build the functional networks for other participants and try to find the general pattern in this functional network.

7.1 Functional Network Inference within Visual Cortex

The visual cortex of the brain is a part of the cerebral cortex that plays an important role in processing visual information. It is located in the occipital lobe in the back of the skull (Which is number 24-30 and 61-64 in figure 4.2). It consists of three region of interests. Here I call it the left visual cortex (24-27), the right visual cortex (61-64) and the central visual cortex (28-30). I choose to apply this method to the visual cortex first because: (1) It contains much less nodes than the whole brain and the changes of its network structure would be easier to observe and interpret. (2) The visual cortex plays an important part in this behavior task and there will definitely be some changes in the functional network structure in the visual cortex.

7.1.1 Inference Results for the Visual Cortex

After applying this method, we have the network density plots in figure 7.1 and the inference results in figure 7.2. We can note from figure 7.1 that the network before the stimulus onset has a higher density than the network after the stimulus onset and is significantly nonzero. The network density after the stimulus onset is lower and has a relatively larger confidence interval.

We can note from the sensor-level networks in figure 7.2 that there are more edges before the stimulus onset and the edges that appear after the stimulus onset also exist in the func-

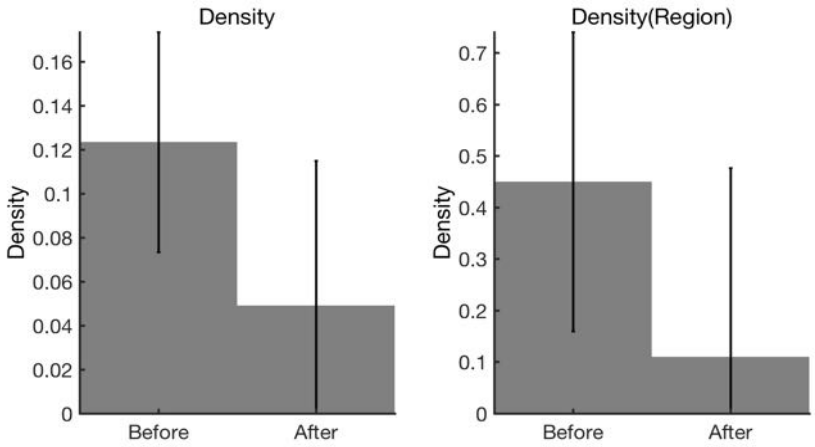

Figure 7.1: This is the figure for the inferred network density, the bar heights represent the average of the network densities, the the whiskers indicate 95% bootstrapped confidence intervals.

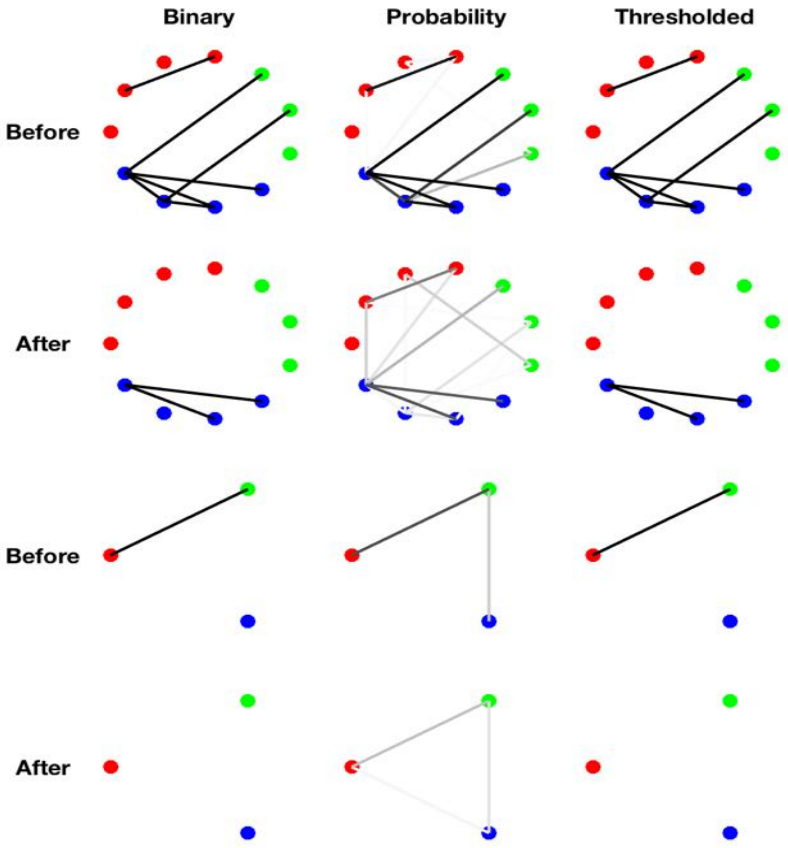

Figure 7.2: This is the inferred functional network within the visual cortex, the red, green, blue nodes correspond to the left, the central and the right visual cortex. The thresholded functional networks are generated by thresholding the probability networks with 0.5, the edges with a probability higher than 0.5 will be kept.

tional network before the stimulus onset. This is a interesting result and I speculate that this is because before the stimulus onset, the participant is in a state of vigilance, so the brain must stay active to keep concentrated, to notice anything he see and process the information he get. This will lead to a lot of functional connectivities in the visual cortex. But after the stimulus onset, the participant has noticed what happened, so many parts of his visual

cortex can relax and he can focus on processing what he saw and analyze the results. This won't lead to many functional connectivities within the visual cortex, which result in fewer edges in the functional network.

The probability functional networks are in the second column in figure 7.2. We can notice that many low-probability edges appear before and after the stimulus onset (especially after the stimulus onset). This is very similar to what we have observed in the simulation chapter, so it's very likely that these light color edges are spurious edges. Then I use 0.5 as a threshold to filter these edges. Edges with a probability higher than 0.5 will be kept, otherwise will be deleted. We can see the results in the last column in figure 7.2. We can note that this network is the same as the binary network in the first column.

However, I find something unexpected in the region-level functional network. Deducing from the sensor-level functional network, there should be no edges after the stimulus onset and an edge linking the central visual cortex (the green node) to the right visual cortex (the blue node). The region-level functional network after the stimulus onset is as excepted (empty), but the functional network before the stimulus onset is different from what I expect. There is only one edge linking the central visual cortex to the left visual cortex. I think this is because that the correlations between the nodes in the central visual cortex and the nodes in the left visual cortex are hard to detect at sensor level but are aggregated and become detectable at region level. While on the other hand, two nodes in the central cortex are connected with two nodes in the right visual cortex, this means that the correlation of these two pairs of nodes in trial is significantly higher than the correlation of these two pairs in the baseline period. However, this not necessarily means that the canonical correlation between the two regions in trial is significant enough to be recognized as an edge, thus no edge is detected in the binary region-level network. From the probability network we can notice that actually the edges between the central visual cortex and right visual cortex exist with a small probability, which suggests some kind of correlation. However, it is not strong enough to be counted as an edge in the binary functional work. We can notice that the edges also show up with a small probability in the region-level probability network after the stimulus onset. And the central-left and central-right correlation seems to be higher than the left-right correlation, but none of them is significant enough to be counted as an edge in the binary network.

Now, I continue to analyze the dynamic functional network for this behavior task. Here I use the same method as used in the simulation chapter. I include 40 time points in each sliding window and proceed with 10 time points overlap. The result is in figure 7.3, we can note that there are some small fluctuations in network density before the stimulus onset and there is an obvious drop in network density at the stimulus onset for both sensor-level network and region-level network. The binary network plots below the density plot also testify this. The time indexes of the binary networks are -90, -60, -30 and 0. We can see that there are some small changes in the binary network before the stimulus onset, however, the network at time 0 is very different, which suggests some big changes in the brain activity.

7.1.2 Inference of Functional Networks for Different Stimulus

One important reason to analyze the functional network is that the structure and the dynamic changes of the functional network is closely related to the real world. Different stimulus can result in different functional networks and different functional networks can also lead to different results and responses. Decoding the relationship between the functional networks and the stimulus can help us a lot in understanding our brain. In this section, I will analyze how functional networks will change in response to different stimulus and the relationship between different functional networks and the experimental results. For this part, I will still

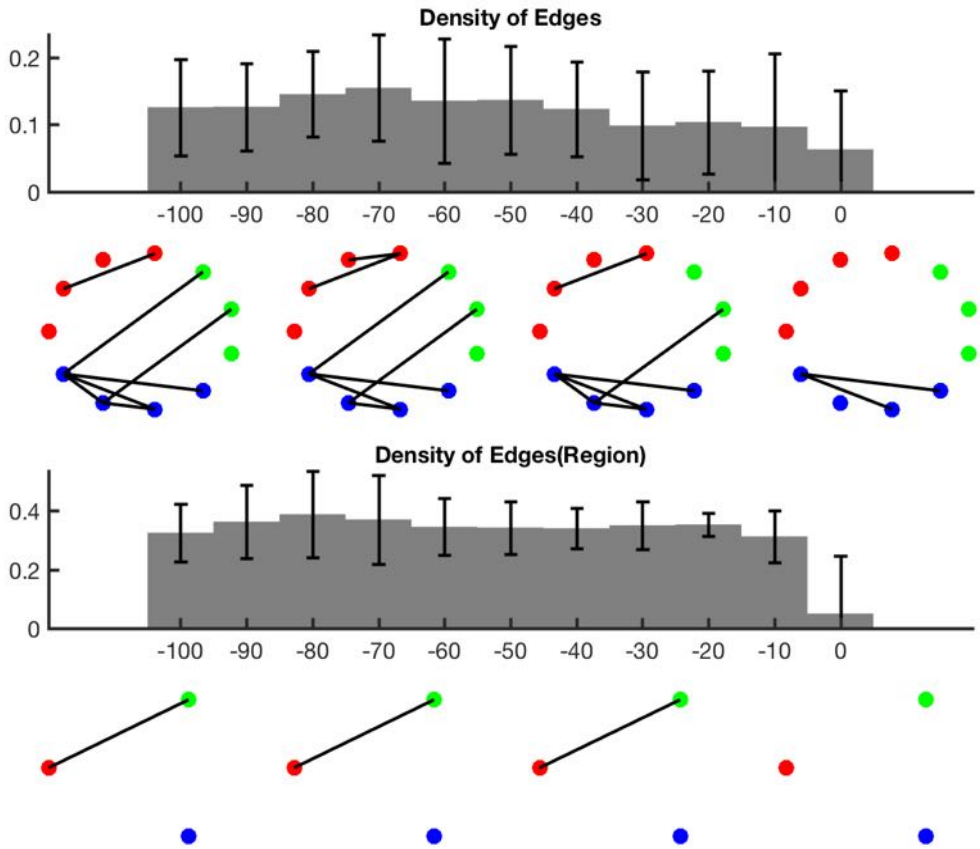

Figure 7.3: This is the dynamic network plot for the EEG data. The time indexes for these binary networks are -90, -60, -30 and 0

keep the analysis within the visual cortex as the changes in the visual cortex will be much easier to observe and analyze.

Left/Right

As introduced in the data chapter, there are many different settings for this experiment, these different settings may have a great influence on the network structure. So first of all, I split the dataset into two parts based on the position of the gap (on the left ring or the right ring) and analyze the functional networks for these two different situations. The result is presented in figure 7.4.

In this figure, the functional networks for the stimulus whose gap is on the left ring are in the first two rows. The last two rows are the functional networks for stimulus whose gap is on the right ring. The three columns in the figure correspond to the binary network, probability network and the thresholded probability network (thresholded by 0.5). We can note that the functional networks for the gap on the right ring and the functional networks for the gap on the left ring are similar before the stimulus onset. This is as expected because there is basically no difference before the stimulus onset for these two settings. We can also notice that the functional networks after the stimulus onset are quite different. I speculate that this is because different positions of the gap on the ring can affect the way the participant see and process the visual information. Thus different parts of the brain will be activated, so the functional networks will be different.

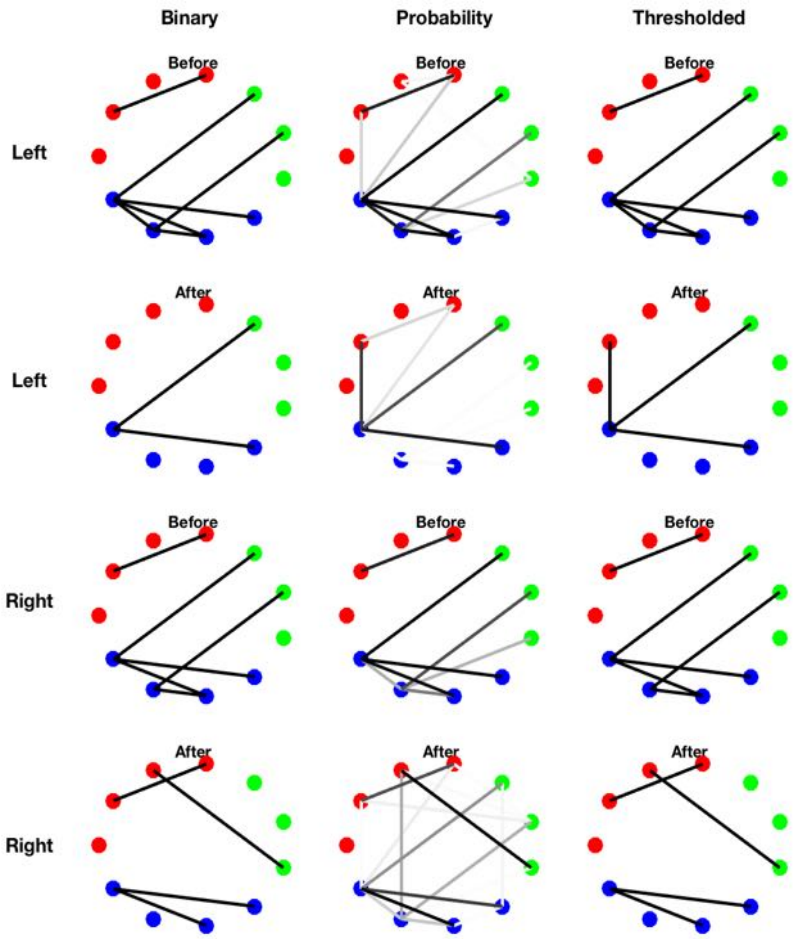

Figure 7.4: In this figure the first two rows are the functional networks for the trials whose gap is on the left ring. The last two rows are the functional networks for the trials whose gap is on the right ring

By looking into these functional networks we can find something more interesting. The red, green and blue nodes correspond to the left visual cortex, the central visual cortex and the right visual cortex. When the stimulus is on the left ring, there is an edge after the stimulus onset connecting the central visual cortex and the right visual cortex. When the stimulus is on the right ring, there will be one edge after the stimulus onset connecting the central visual cortex and the left visual cortex. As we know, the right part of our body is mainly controlled by our left brain and the left part of our body is mainly controlled by our right brain. So I speculate that if the stimulus is put on the left ring, our left eye will be more activated (i.e., the eye balls will move to the left, and the left eye will tend to see the gap more clearly and obtain more information) and thus our right visual cortex will be more active and have more functional connectivities with the central cortex. This is the same if the gap is on the right ring, thus there will be an edge linking the left visual cortex and the central visual cortex.

Correct/Wrong

As introduced before, the accuracies of the answers are recorded and are represented by number 0,1,2,3,4,9. Because the accuracy in a large extent is determined by how the participant perceive and process the visual stimulus. So the networks corresponding to different

results (here I focus on whether the participant gave the right answer, meaning whether the accuracy is 0 or not) may be different.

We notice that in this dataset, there are 200 trials that the participant gave the correct answer and in the other 55 of the trials the participant gave the wrong answer. This result don't seem to be random, so it's reasonable to speculate that there are some differences in the process of perceiving and processing the signals so sometimes the answer is right and sometimes the answer is wrong. With this speculation, I'm going to build the functional network by using the trials with the correct answer and the trials with a wrong answer. So I split the dataset into two parts based on whether the accuracy is 0 or not and apply this inference procedure to these two datasets, the result is in figure 7.5.

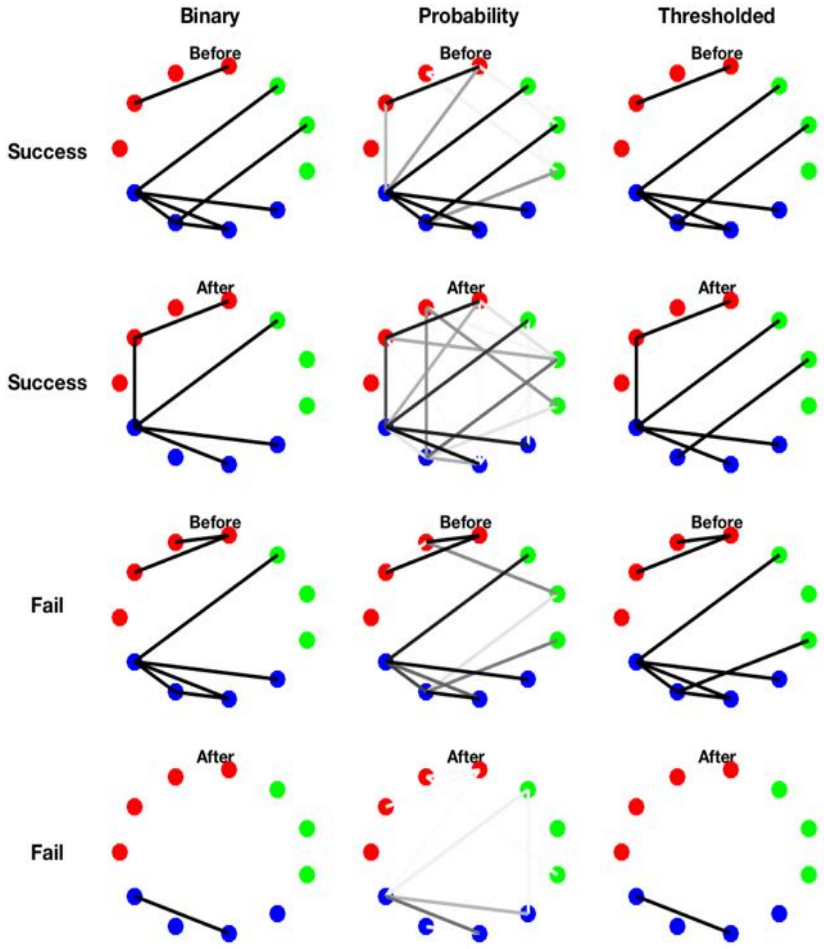

Figure 7.5: In this figure the first two rows are the functional networks for the trials in which the participant gave the right answer. The last two rows are the functional networks for the trials in which the participant gave the wrong answer or didn't give any answer.

We can note in figure 7.5 the functional networks inferred from the two datasets. The "y label", success and fail, represents whether the participant give the correct answer. We can note from this figure that the functional networks before and after the stimulus onset are quite different, especially after the stimulus onset. The binary network and the thresholded network after the stimulus onset for the success trials contain 5,6 edges. However, only one edge show up in the "fail" binary functional network and thresholded functional network. This result suggests that after the stimulus onset, there is no strong functional connectivity in the visual cortex for this participant, he may somehow missed the gap and couldn't process

the information. This is a good explanation why the participant gave the wrong answer (or didn't give any answer) in these trials.

7.2 Functional Network for the Whole Brain

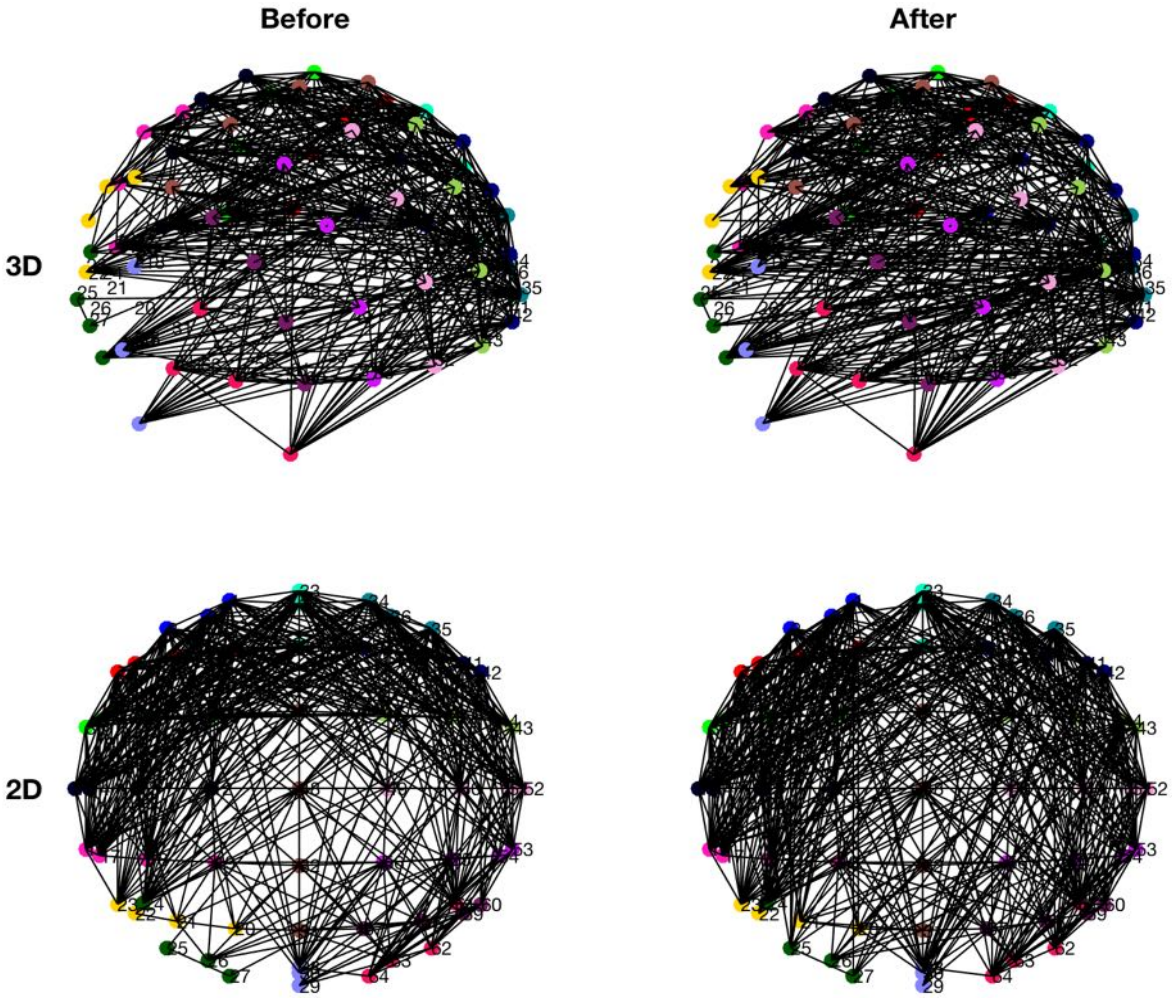

Figure 7.6: The sensor-level functional networks for the whole brain. The nodes are color-coded by region of interests defined in section 4.2.

After applying this inference procedure to data of the visual cortex, we can get some idea of how this method is behaving on real data. Then I continue this exploration and build the functional network for the whole brain, the result is in figure 7.6. The first and second row contain the 3D and 2D plot of the functional network. The first and second column correspond to before the stimulus onset and after the stimulus onset. All the nodes in this figure are color coded by different regions as defined in figure 4.2. Because the edges are too dense, it is really difficult to find the specific changes from after the stimulus onset to before the stimulus onset. However, we can note that the functional network in the visual cortex region is more sparse than the other regions (especially before the stimulus onset). And after the stimulus onset, there are more edges linking the visual cortex to other parts of the brain compared to the functional network before the stimulus onset. I speculate that this is because more brain activities related to the transition of the information, the processing of the signals will happen after the stimulus onset. So there will be more functional connectivity

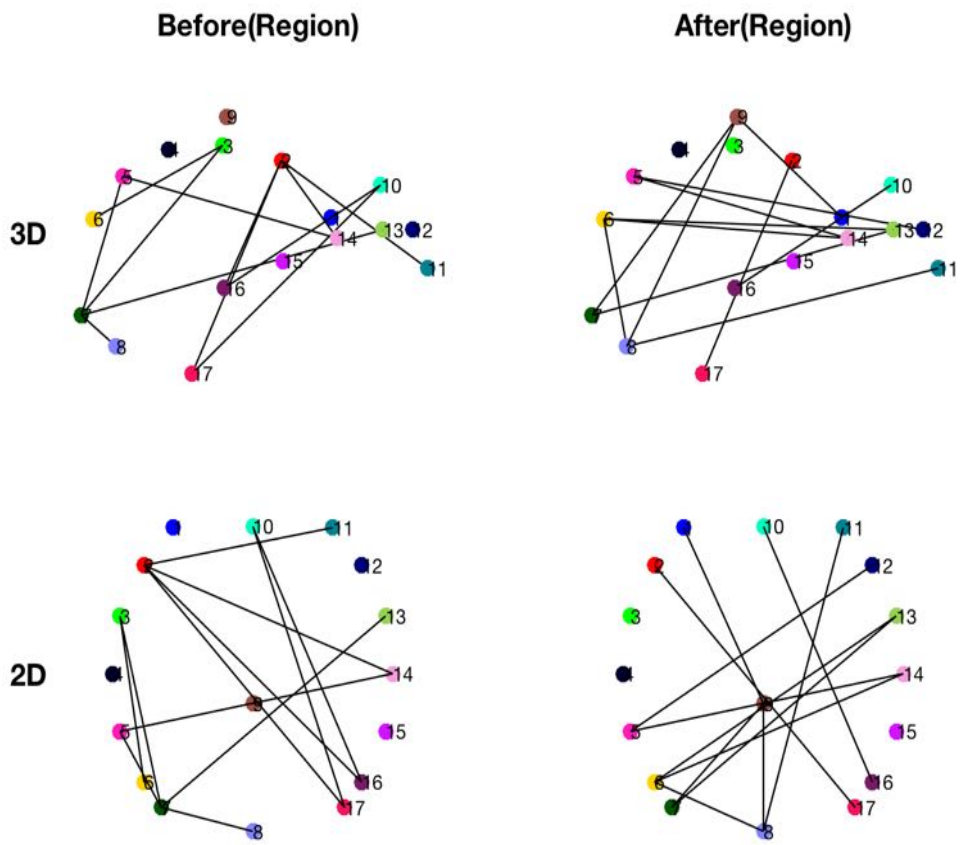

Figure 7.7: The region-level functional networks for the whole brain. There are 17 regions in this network and each region is represented by one color.

to help transfer the visual information. From the density plot in figure 7.8 we can notice that there is a slight increase in network density after the stimulus onset, however, it is not statistically significant.

The region-level functional networks are presented in figure 7.7. There are 17 regions in this figure, each region is represented by a node with a unique color as the same as used in the sensor-level network. Node locations in the region-level functional network are the average locations of the electrodes corresponding to the ROI. We can note that, as illustrated before, there are more edges linking the central visual cortex to other regions of the brain after the stimulus onset, this is the same as shown in the sensor-level functional network. And with fewer edges and nodes, this is more obvious in the region-level network. The density plot in figure 7.8 also suggests that the network density after the stimulus onset is higher than the network density before the stimulus onset, these patterns are consistent with the sensor-level functional network.

7.3 Results for Different Datasets

The inference result for the dataset 14 is a good start. Now I want to continue this journey to build the functional networks for other participants and try to find the similar patterns in their functional networks. This is a difficult task as between-subject variations are very common for experiments like this. The scientists have even found the functional networks' correlation with a variety of individual measures (e.g., IQ, personality). Besides, different locations of the electrodes and hardware instability can also introduce unexpected influence

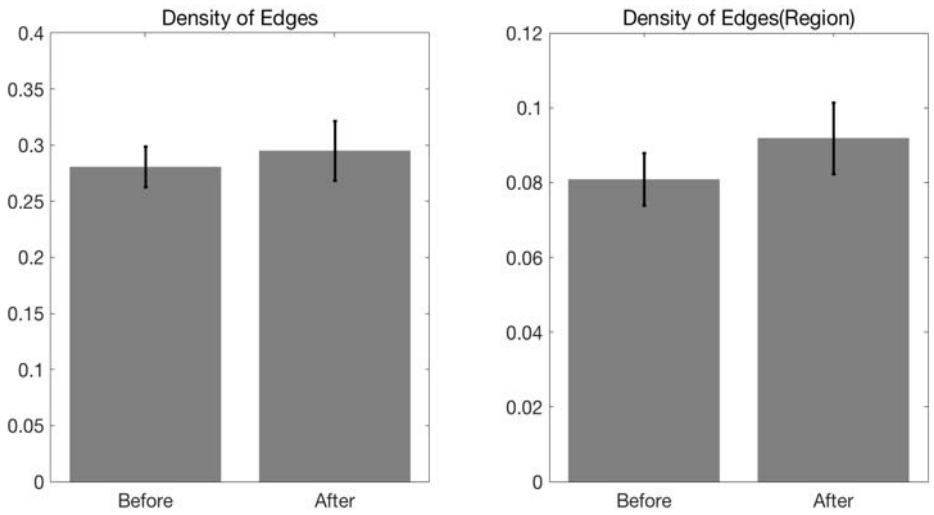

Figure 7.8: The network density

to the functional network. They can make the real patterns really hard to find.

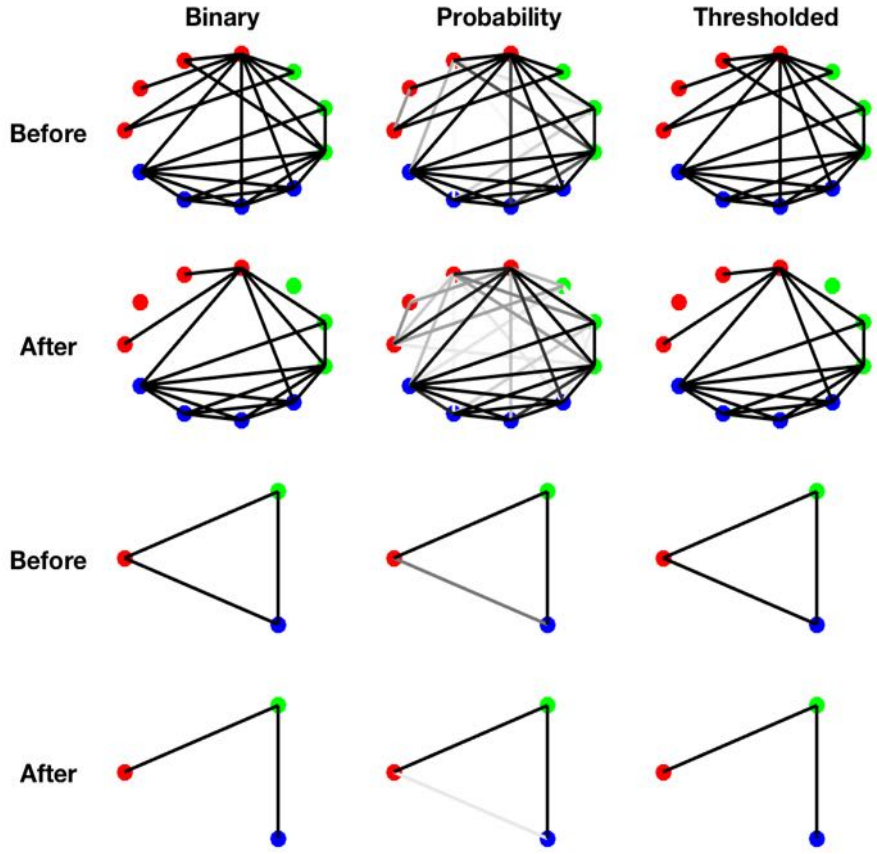

Figure 7.9: The functional network for dataset 1 (Visual Cortex)

However, I still continue this exploration. Figure 7.9 presents the functional networks in the visual cortex for dataset 1. In this figure we can note that in the sensor-level functional network, there are 23 edges before the stimulus onset and 18 edges after the stimulus onset. This is much more than the edges in the pervious network (which is 7 before the stimulus

onset and 2 after the stimulus onset). By taking a closer look we can find that the edges that appear after the stimulus onset also exists before the stimulus onset, this is the same as we have observed before. The region-level functional network also testifies this, we can note that before the stimulus onset there are 3 edges linking all the three regions. However, after the stimulus onset, the edge between left visual cortex and right visual cortex disappears, I speculate that this is because after the stimulus onset the participant will focus more on processing the visual information, which need more functional connectivity between the central visual cortex and the left/right visual cortex, thus the edge linking the left visual cortex and the right visual cortex will no longer persist. And we can notice from the probability network that even before the stimulus onset, the probability of the left-right edge is the lowest one.

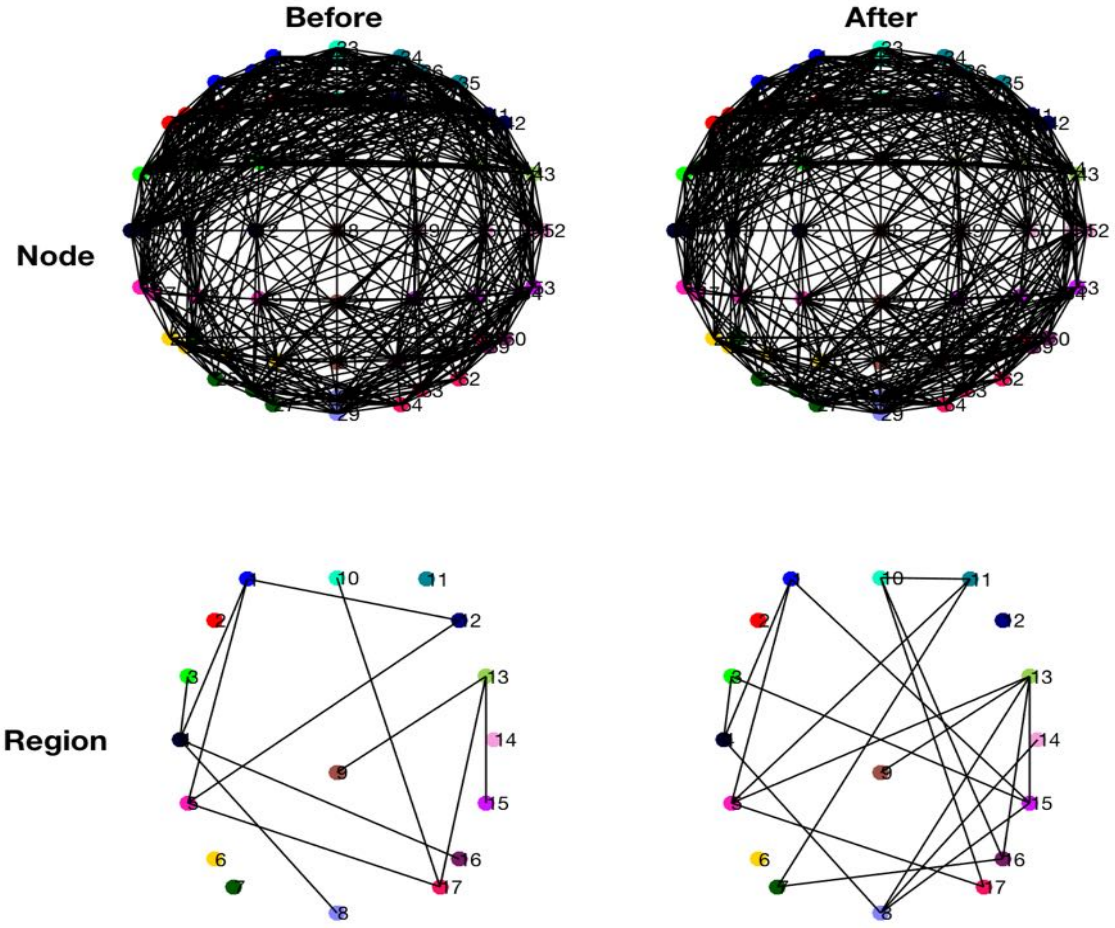

Figure 7.10: The functional network for dataset 1

Figure 7.10 is the functional network for dataset 1 for the whole brain. It's easy to notice that in the sensor-level functional network, the edges are more dense in the frontal cortex before the stimulus onset. And also, there are more edges linking the visual cortex to other parts of the brain after the stimulus onset, this is also verified by the region-level functional network. These patterns are the same as the functional network we have shown for dataset 14. Besides, in both the sensor-level network and region-level network, the network density after the stimulus onset is higher than the network density before the stimulus onset. The density plot of this figure is very similar to figure 7.8 and is not included here.

Figure 7.11 presents the function networks within the visual cortex for dataset 7. We

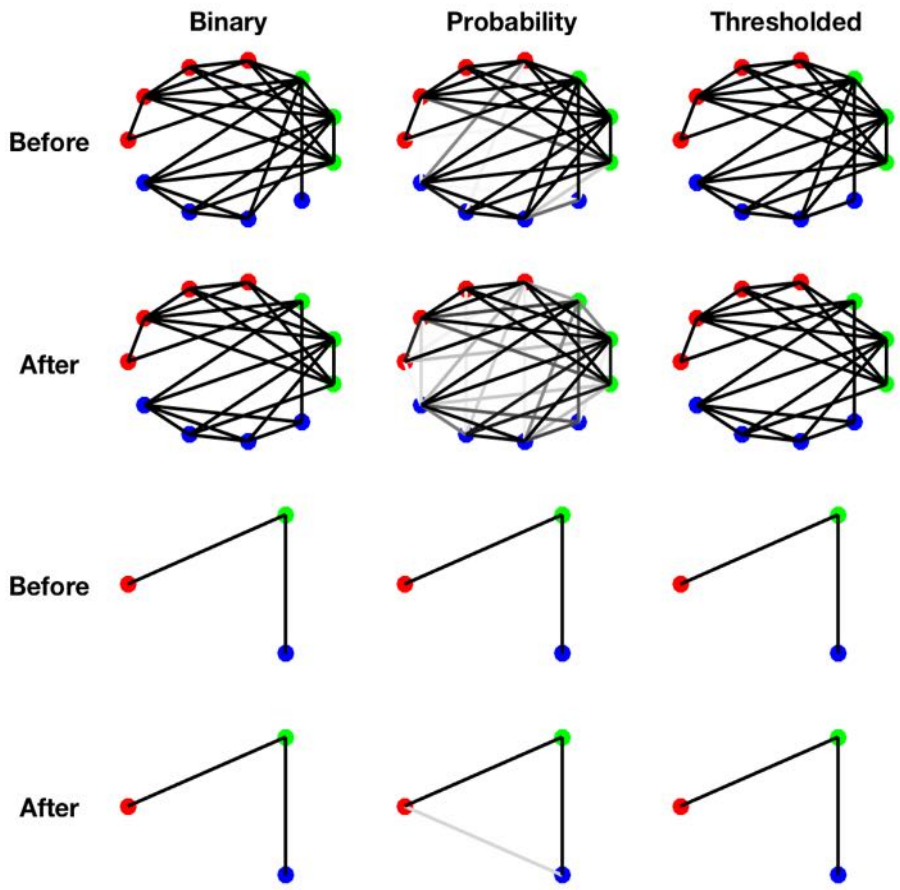

Figure 7.11: Functional Network for Dataset 7 (Visual Cortex)

can see that there are 27 edges before the stimulus onset and 23 edges after the stimulus onset. Nearly all of the edges after the stimulus onset are in the functional network before the stimulus onset. These functional networks are pretty similar to the functional networks in figure 7.9. We can note from this figure that there are more central-left and central-right connectivities than the left-right connectivities. This is also verified by the region-level functional network in the last two rows in this figure.

The functional network for dataset 7 for the whole brain is shown in figure 7.12. We can see that: (1) there are more edges in the frontal cortex. (2) There will be more edges linking the visual cortex to other parts of the brain after the stimulus onset. These are the same for the first two function networks.

So from these experiments, I want to draw a conclusion that, in this behavior task, before the stimulus onset, the visual cortex will be more active to note the stimulus, and there is not so much information to transfer from the visual cortex to other parts of the brain. As a result, there will be relatively more edges within the visual cortex and less edges linking the other parts of the brain. While after the stimulus onset, there will be more brain activities to process the visual stimulus, so there will be more functional connectivities between the visual cortex and other parts of the brain. The specific changes for the functional network within the visual cortex is hard to generalize. However, I think we can expect more ”central-left” and ”central-right” connections.

Although this method seems very promising in the simulation part, the results of practice

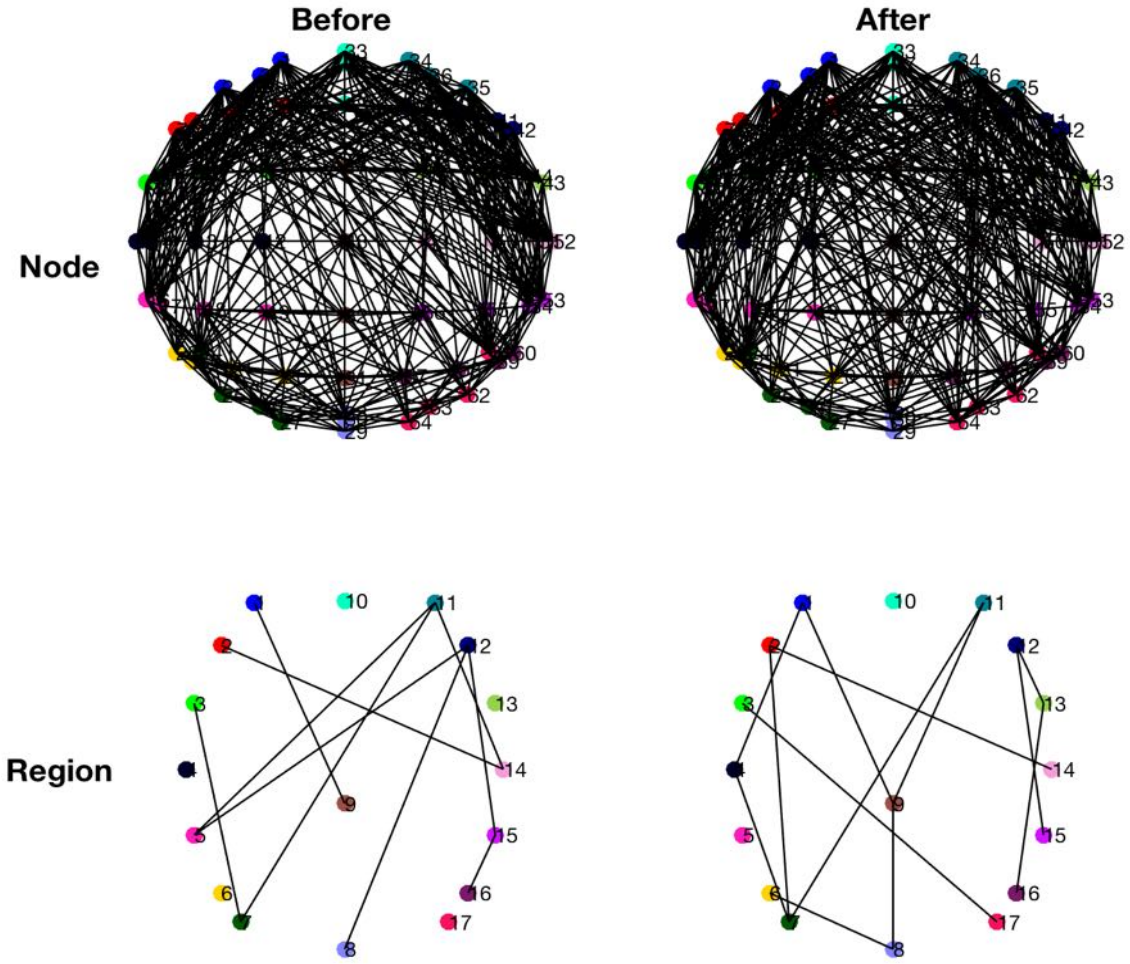

Figure 7.12: Functional Network for Dataset 7

suggest that the real functional network is still very hard to find and generalize. I think there are three reasons:

1. The true functional networks for different people are different. As described above, the functional network is even correlated with IQ and personality.

2. The positions of the electrodes are different, sometimes the corresponding electrodes for different participants may record the brain activities for different brain regions.

3. The method I use can also bring a lot of uncertainty. When I build the functional network, I am not only analyzing the data during the trial. However, I compare the time points in the trial with the baseline data, this can introduce a lot of uncertainty in the result. Even if the brain activities during the trials are the same for two participants, the inference results can be different given different baseline data.

One last thing I want to specify here is the choice of the dataset that is presented in the project. In this project, I gave a most detailed description of the inference procedure of dataset 14, this is because (1) its functional network for the whole brain is normal and representative, and (2) it has the least edges in the visual cortex, these edges also appear in most of the other functional networks, this means that these edges are highly possible to be in the real functional network and are worth investigation. What's more, fewer edges means a smaller chance to have false positives and also means a simpler network structure, which makes the changes more observable and easier to interpret.

Chapter 8

Discussion

8.1 Sensor-level Network VS Region-level Network

After all the analysis and building the sensor-level and region-level functional network, we can note that these two functional networks are actually very different. Each network has its advantages and disadvantages. If we want to figure out the details, the specific changes of the network, the region-level functional network is a better choice. Compared to sensor-level functional network, it is more robust, stable and easier to interpret. For example, in the analysis within the visual cortex, we can note that the it's really difficult to understand and interpret the changes of the sensor-level network. There are two reasons: (1) There are too many nodes and edges in the network, it's almost impossible to understand the specific meaning of each edge and node. (2) Biological data is very unstable and can carry a lot of noise, due to limited sample size, sensor-level networks are sensitive and susceptible to noise and outliers, so it's really difficult to extract reliable details from sensor-level networks. On the other hand, the region-level functional network is more reliable in details, it aggregates the data from many sensors, thus it can make use of more data points and has a stronger statistical power. So we can note from the simulation results that the region-level networks are more stable and robust, fewer false positives will show up and the true edges always have high probability of appearance. From the analysis of EEG data we can also notice stronger central-left and central-right correlation. This is very difficult to conclude if we only have sensor-level network. Besides, it's also easier to interpret the specific changes in region-level networks, because the predefined regions usually have their specific functionalities, so it would be easier to understand and interpret why the active regions in the brain changes.

Another advantage of region-level network is that it will be very easy for them to perform across-subject analyses. For example, in EEG recordings, the electrode locations is usually different for each subject, so it's difficult to compare the sensor-level functional network for different perticipant. However, the regions the sensors cover are often the same. So if we use the same ROIs to construct region-level networks, it would be much easier to compare between different subjects. In addition, the data for each ROI for different participants can be aggregated together before building the functional network. Thus we can estimate the across-subject networks. These across-subject networks could detect the connectivities that are not subject-specific. What's more, as introduced before, region-level functional network can detect the edges that can't be detected by sensor-level network. Sometimes the correlation between sensor-level networks is too weak to detect, but the correlation can be aggregated and strengthen by this region-level network inference method.

However, one shortcoming of canonical correlation is that its performance can be affected by the number of signals (electrodes) in the groups (ROIs) being compared. Because

more signals can give more degrees of freedom for the linear combinations of calculating the canonical correlation. Thus the canonical correlation values tend to be higher when we calculate it between ROIs with more component electrodes. It doesn't mean that it would be easier to detect an edge between regions with more electrodes, on the contrary, it would be more difficult to detect an edge between regions with more nodes. This is because in this inference procedure, the canonical correlation value in the trial between two ROIs is compared to a baseline distribution calculated on the same two ROIs. If there are many nodes in the regions, the canonical correlation in the baseline can be very high, so it would be very difficult for the canonical correlation during trial to be consistently higher. So we should control the number of electrodes in the ROIs to avoid this phenomenon. According to my ROI definition, each ROI contains 3-4 electrodes.

Sensor-level networks also have their own advantages, especially when it comes to the network for the whole brain. When we view the sensor-level network from a macroscopic scale, things can be very interesting. We can note that for some parts of the brain, the edges are really dense while for other parts, there are very few edges. This phenomenon is notable in sensor-level network, but not so obvious for region-level network. This is because the Benjamini-Hochberg procedure is very "strict" in detecting the edges, which can lead to many false negatives. There are not so many potential edges in the region-level network, so these false negatives may greatly affect the structure of the network, which can blur the real pattern of the network. However, for sensor-level network, there are many potential edges, it's very difficult for the appearance of false negatives to change the pattern of the sensor-level functional network (It's similar to the swarm intelligence). So it would be easier for us to notice the change of the network pattern, which brain regions are more active.

8.2 Robustness to Persistent Correlations

Constant correlations due to recording apparatus and brain activities unrelated to the experiment can be really bothering and may affect the inference result of the functional network. However, one advantage of this method is that it is insusceptible to systematic spatial regularities in the correlations as long as it also exists in the baseline data. This is because the edges come from the comparisons: the correlations during the behavior task must be stronger than the correlations during the baseline periods in order to achieve significance. As long as the correlation exists in both trial and baseline, it won't be counted as an edge. Actually we can note that the background correlation exists in all nodes during the entire trial is helpful in my inference as they cause the baseline correlations to be higher and make it more difficult for spurious correlations to appear as false positives. However, this inference method is not immune to all the noises. The correlations due to deep subcortical sources of electrical activity can affect the inference result of the functional networks. In this experiment, deep subcortical source not present before the stimulus onset that emerges during the task could bring correlations after the stimulus onset that would be inferred by this method as edges. These edges are spurious, because they don't represent correlated brain activity appearing locally at each sensor. However, as introduced before, EEG recording is extremely uninvasive and insensitive to distant electrical sources. So the electrical activity from deep subcortical source won't be a very big problem in this analysis.

8.3 Multiple Comparisons

Multiple comparisons is an important issue in this project. Because we need to take each pair of electrodes into account, so this 64-electrode network (typical in EEG recordings) means 2016 statistical tests. Here, I deal with multiple comparisons using the Benjamini-Hochberg

procedure. This procedure is very good in controlling the proportion of false positive edges relative to the total number of detected edges, however, it addresses the connections that become extreme compared to the baseline period: the p-values for the edge needs to be very small to be counted. As a result, this method is not so convincing in explaining the absence of the edges: if an edge exists in an inferred network it means that there is a strong difference between trial and baseline coupling, but if an edge does not exist, there might also be a connection but just was not strong enough to be classified as significant. This could be a big problem if the SNR of the signal is low. Then it will be very difficult for the nodes achieve low p-values in the trial period. So this may result in many false negative edges. Besides, the power of the Benjamini-Hochberg procedure is different for networks with different nodes, by comparing figure 7.9 with figure 7.10 and figure 7.11 with figure 7.12, we can note something interesting. The red, green and blue nodes in figure 7.9 and 7.11 corresponds to nodes with number 7, 8 and 17 in figure 7.10 and figure 7.12. The network between then are inferred from the same data using the same method, however, we can note that the connection is different. Within these three nodes, there are more edges in figure 7.9 and figure 7.11 than figure 7.10 and figure 7.12. This is because there are more potential edges in figure 7.10 and figure 7.12, thus the p-value for the edges in figure 7.10 and figure 7.12 need to be much lower than the p-value for the same edges in figure 7.9 and figure 7.11 to achieve significance.

Another thing that concerns the Benjamini-Hochberg procedure is that it sets a lower bound on the number of edges to be detected. According to this method, the smallest possible p-value, $p_{(1)}$, is bounded below by the number of times I resample in the bootstrap procedure (here 1/1000). In a typical network, there may be many pairs of edges whose test statistics is higher than all the test statistics calculated from the baseline period, leading to a set of p-values $p_{(1)}$, ..., $p_{(k_{min})}$ equal to 1/1000. If the total number of these, k_{min}, satisfies

$$p_{(kmin)} = \frac{1}{1000} \leq \frac{qk_{min}}{N_{MC}}$$

then all of the tests 1,..., k_{min} will be considered significant and there will be an edge for all of them. But on the other hand, if $p_{(k_{min})} \geq \frac{qk_{min}}{N_{MC}}$, then none of the tests will be reported significant. As a result, the inferred network will be reported to be empty. From this we can note that there is a lower bound on the number of edges that can be detected: the detected network will contain edges more than or equal to the lower bound or be reported as empty. From the above equation we can know that the smallest number of edges that can be detected is:

$$\frac{N_{MC}}{qN_{bs}}$$

where N_{bs} is the number of bootstrap samples used for estimating of the null distributions. And we can also note that N_{MC} in this formula equals to $\frac{N\times(N-1)}{2}$, where N is the number of electrodes. So the lower bound can be really large when the size of the networks is big, this can a big problem for our analysis.

There are some possible ways to deal with this. First of all, we can reduce the number of nodes in the networks. This can be done by defining the networks on a larger spatial scale such as region-level network. With fewer possible edges, from the formula we can know that the lower bound of the edges will also decrease. Another approach is to increase the number of bootstrap samples to reduce the minimum number of detectable edges. This is a reasonable method, since most of the time the networks will have more enough edges to exceed the threshold. If the resulting inferred networks are not empty, that's a great news and we can take it as our result. If the networks are still empty, then, there are two possibilities:

(1) there is no correlation between the nodes. (2) the number of true edges is smaller than the detectable amount. In this case, two other options are available: (1) estimate the null distributions parametrically. (2) estimate the tails of the null distributions above the highest bootstrap sample using extreme value theory (Emily P. Stephen et al., 2014).

8.4 Data Preprocessing

Usually, the EEG data used in analysis needs to be filtered by a high-pass filter and a low-pass filter. The high-pass filter is typically set to 0.5-1 Hz and the low-pass filter is usually set to 35-70 Hz. The high-pass filter is meant to remove slow artifact, such as electrogalvanic signals and movement artifact. The low-pass filter is used to remove high-frequency artifacts, such as electromyographic signals.

When I start to build the functional network, I filtered the data. However, the result is really bad. Very few edges were detected and some of the inferred networks are even empty. Besides, the dataset I use is the cleaned version. Many of the artifacts in the data have already been removed. Filtering it may lead to over-processing and the loss of information. So in this project, I used the data directly without filtering it.

Bibliography

[1] Stephen E. P., Lepage K. Q., Eden U. T., Brunner P., Schalk G., Brumberg J. S., et al. (2014). *Assessing dynamics, spatial scale, and uncertainty in task-related brain network analyses.*, Front.

[2] Sui J., Adali T., Yu Q., Chen J., Calhoun V. D.(2012). *A review of multivariate methods for multimodal fusion of brain imaging data.*, J. Neurosci. Methods.

[3] R.M. Hutchison, T. Womelsdorf, E.A. Allen, P.A. Bandettini, V.D. Calhoun, M. Corbetta, S. Della Penna, J.H. Duyn, G.H. Glover, J. Gonzalez-Castillo, et al. (2013) *Dynamic functional connectivity: promise, issues, and interpretations*, Neuroimage.

[4] Bullmore, E., and Sporns, O. (2009) *Complex brain networks: graph theoretical analysis of structural and functional systems.*, Nat. Rev. Neurosci.

[5] Kramer, M. A., Eden, U. T., Cash, S. S., and Kolaczyk, E. D. (2009). *Network inference with confidence from multivariate time series.*, Phys. Rev. E Stat. Nonlin. Soft Matter Phys.

[6] Sakoglu U, Pearlson GD, Kiehl KA, Wang Y, Michael A, Calhoun VD. (2010) *A method for evaluating dynamic functional network connectivity and task-modulation: application to schizophrenia.*, MAGMA.

[7] Buckner RL, Sepulcre J, Talukdar T, Krienen FM, Liu H, Hedden T, Andrews-Hanna JR, Sperling RA, Johnson KA. (2009) *Cortical hubs revealed by intrinsic functional connectivity: mapping, assessment of stability, and relation to Alzheimer?s disease.*,J Neurosci.

[8] Kiviniemi V, Vire T, Remes J, Elseoud AA, Starck T, Tervonen O, Nikkinen J. (2011) *A sliding time-window ICA reveals spatial variability of the default mode network in time.*, Brain Connect.

[9] Dalton KM, Kalin NH, Grist TM, Davidson RJ. (2009) *Neural-cardiac coupling in threat-evoked anxiety.*. J Cogn Neurosci.

[10] Neuroscience. (n.d.). In Wikipedia. Retrieved June 1, 2017, from `https://en.wikipedia.org/wiki/Neuroscience`

[11] Biological Neural Network. (n.d.). In Wikipedia. Retrieved June 1, 2017, from `https://en.wikipedia.org/wiki/Biological_neural_network`

[12] Landolt C. (n.d.). In Wikipedia. Retrieved June 1, 2017, from `https://en.wikipedia.org/wiki/Landolt_C`

[13] Electroencephalography. (n.d.). In Wikipedia. Retrieved June 1, 2017, from `https://en.wikipedia.org/wiki/Electroencephalography`

[14] Independent component analysis. (n.d.). In Wikipedia. Retrieved June 1, 2017, from `https://en.wikipedia.org/wiki/Independent_component_analysis`

[15] Time-frequency analysis. (n.d.). In Wikipedia. Retrieved June 1, 2017, from https://en.wikipedia.org/wiki/Time-frequency_analysis

Appendices

Appendix A

More Figures and Analysis on Other Samples

In the previous chapters we have analyzed and shown the functional networks for three datasets. They are quite typical and are really helpful for us to figure out what a normal functional network is like for this behavior task. Here in this appendix, I want to show more functional networks for other participants. Besides showing the typical and normal ones, I will also present some of the functional networks that looks ”abnormal”. They can provide a more complete view for the experiment results and can give more information about the functional network.

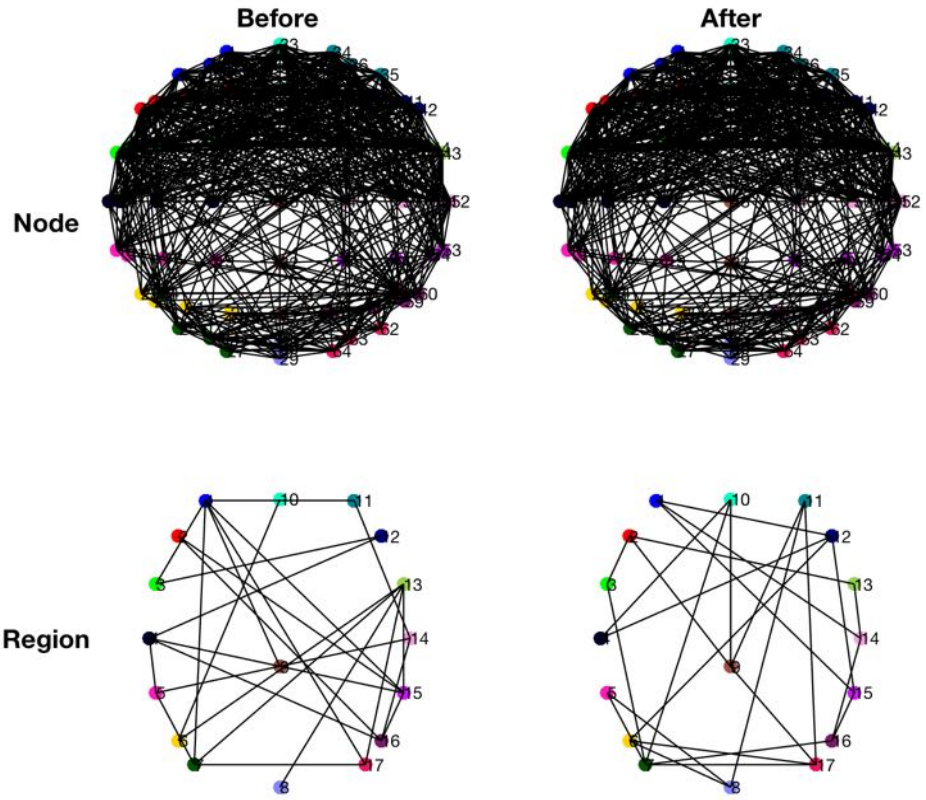

Figure A.1: Functional Network for Dataset 15. This functional network is similar to most of the functional networks, there are more edges in the prefrontal cortex and after the stimulus onset there will be more edges linking the central visual cortex to other regions of the brain

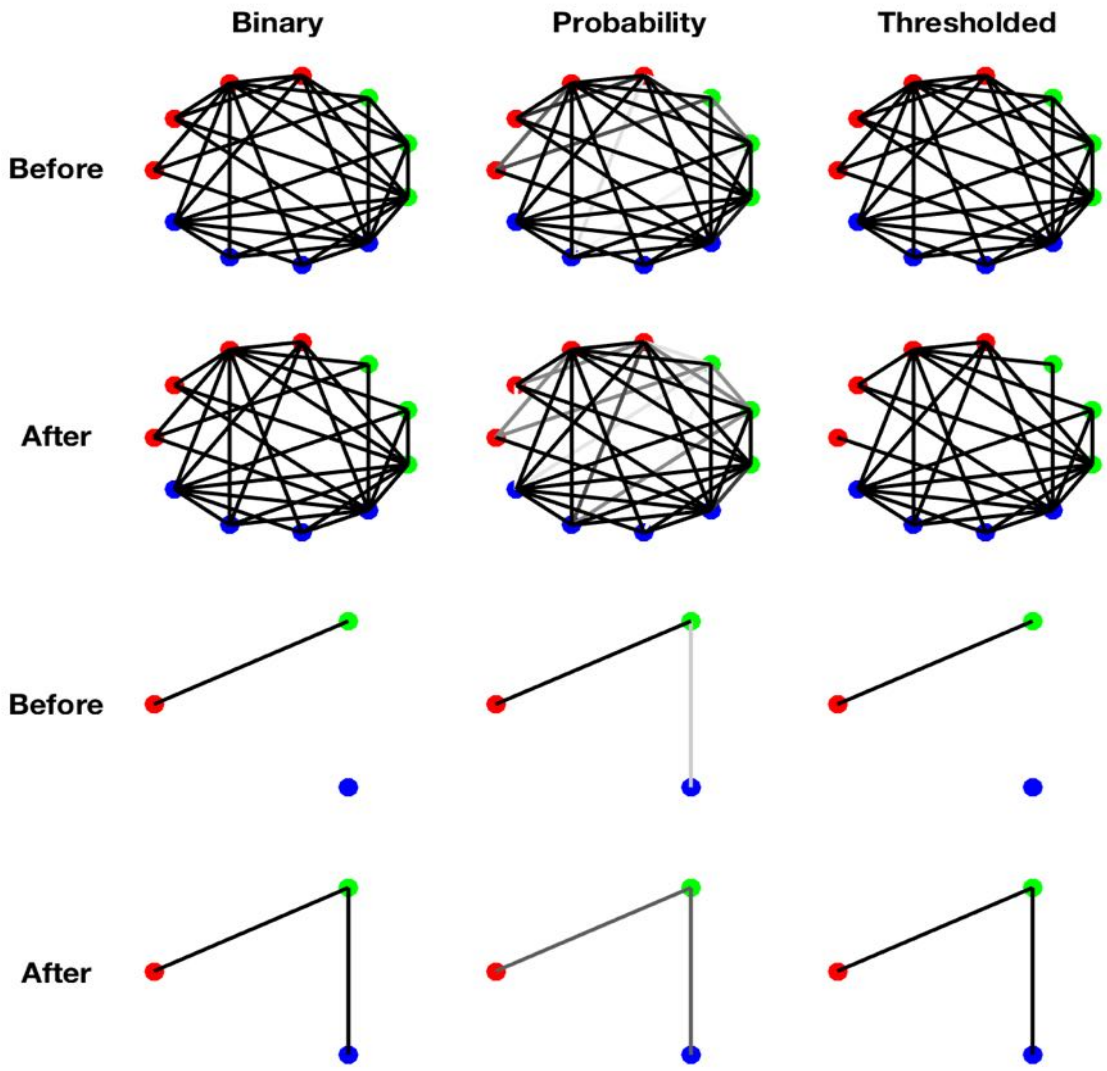

Figure A.2: Functional Network for Dataset 15 (Visual Cortex). There are many edges in the sensor-level functional network (32 before the stimulus onset and 29 after the stimulus onset). In this figure the region-level functional is more simple and clear. It's much easier for us to notice stronger central-left and central-right connections from the region-level functional network.

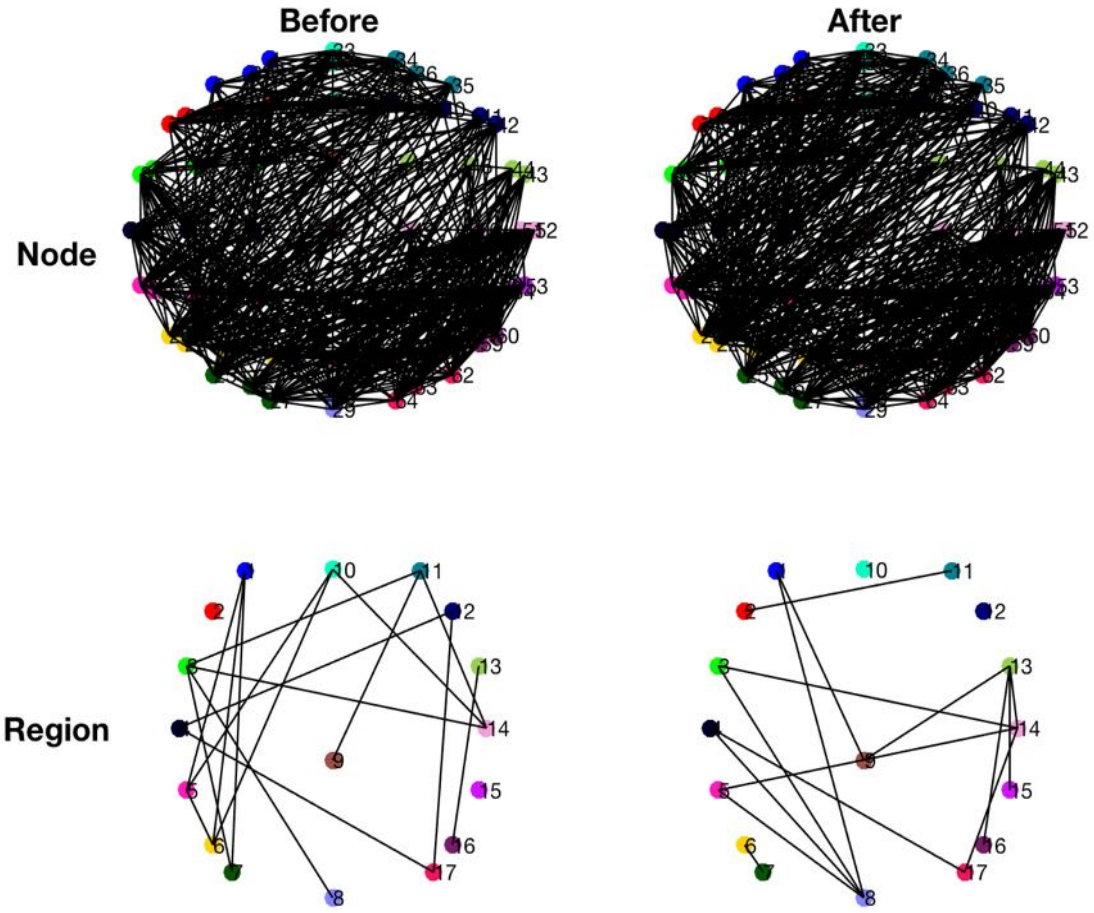

Figure A.3: Functional Network for Dataset 5. This functional network is not so typical, it has the highest network density among all the 16 functional networks (0.337 before the stimulus onset and 0.365 after the stimulus onset). I speculate that this is because in the baseline period the correlation between the nodes are too small. We can also note that there are more connectioins in his left brain, but I don't know the reason for it.

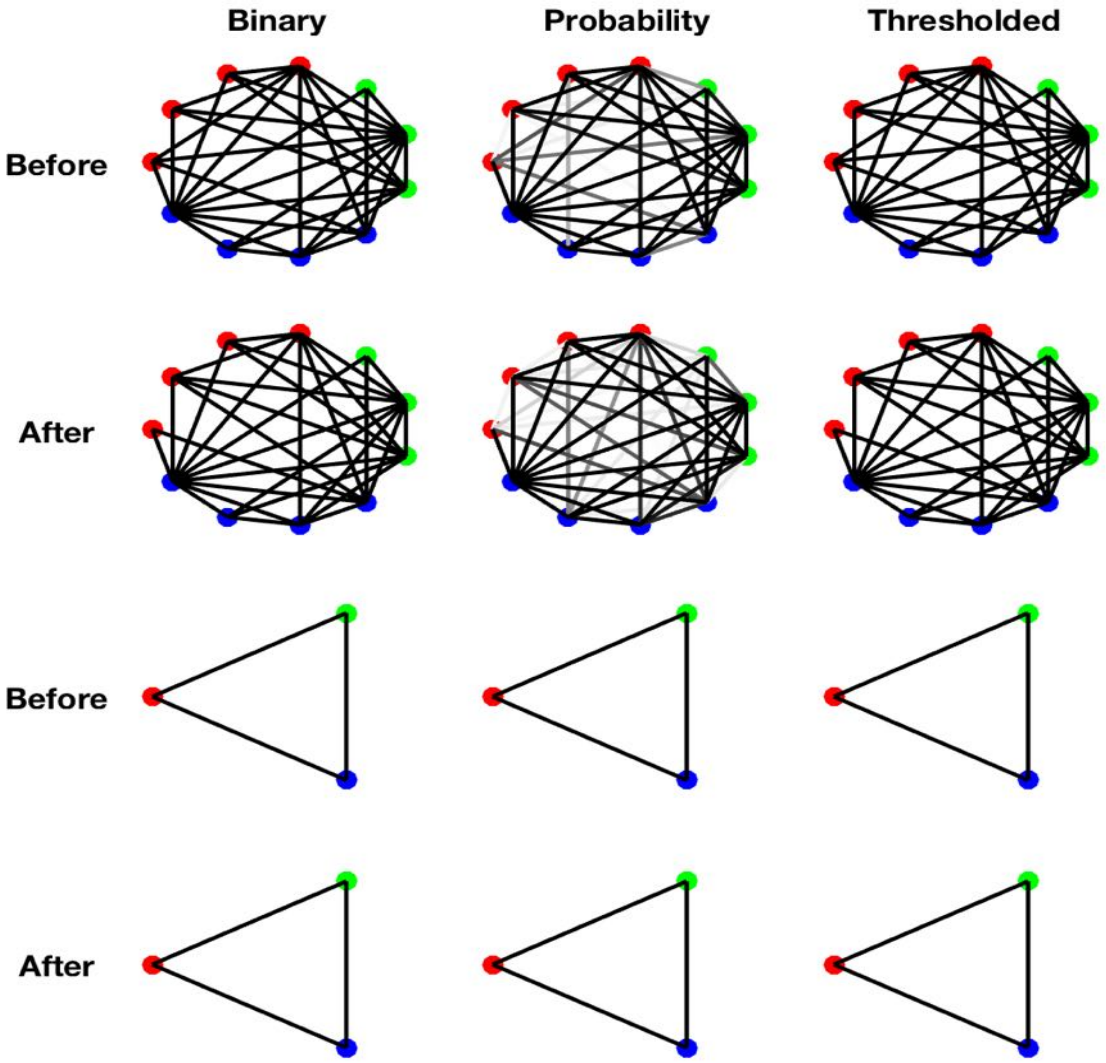

Figure A.4: Functional Network for Dataset 5 (Visual Cortex). As expected, this network also has the most edges among all the networks (34 before the stimulus onset and 31 after the stimulus onset). Although it's really difficult to analyze the specific changes in network structure, we can note that the network density still decreases after the stimulus onset and the edges that appear after the stimulus onset all appear before the stimulus onset, this is the same as almost all the functional networks we have before. The region-level functional network is a complete graph. Because there are too many edges, there will also tend to be many spurious edges, this makes it very difficult to find the real functional network.

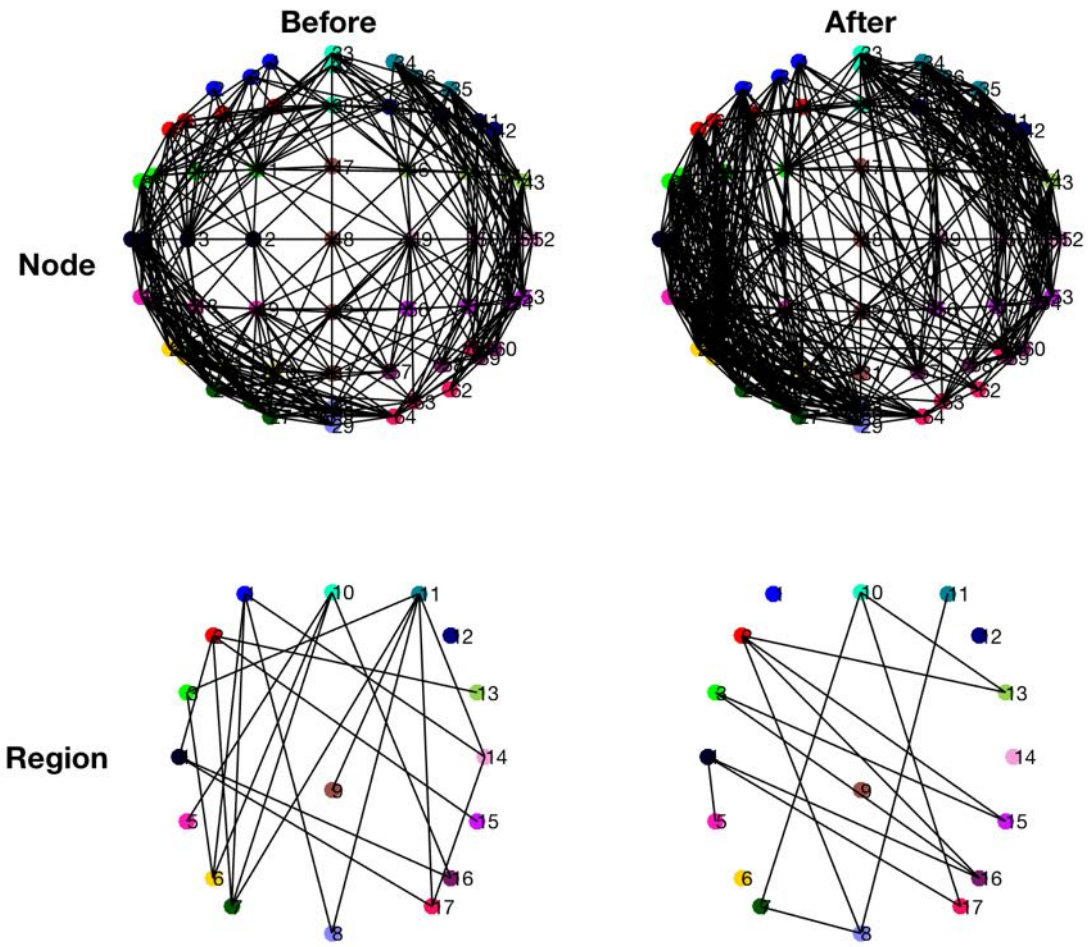

Figure A.5: Functional Network for Dataset 3. This functional network is another nontypical example. Its density is relatively low (0.221 before the stimulus onset and 0.283 after the stimulus onset). The structure of the sensor-level network is also strange, it has many edges in the topright corner and bottomleft corner, while there are very few edges in the other regions.

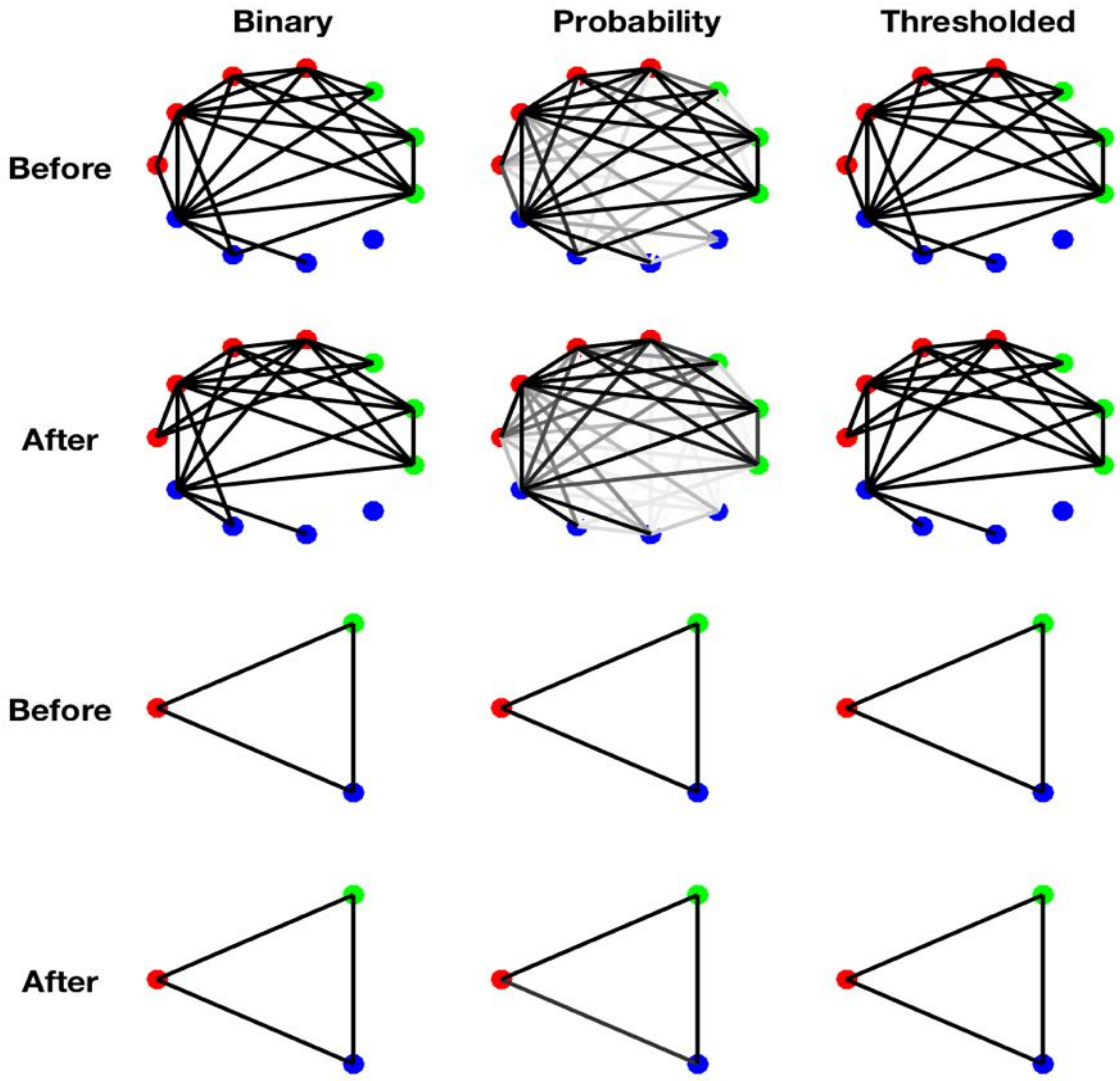

Figure A.6: Functional Network for Dataset 5 (Visual Cortex). The sensor-level functional network is also not typical, there are many edges in the regions of left visual cortex and central visual cortex. However, very few edges show up in the right visual cortex. By comparing this functional network with the functional network of the whole brain we can find the answer: the right visual cortex is in the bottomright corner in the scalp, where there are very few edges. We can also note that although there is a big increase in network density for the whole brain, the number of edges in the functional network of the visual cortex still decreases (24 edges before the stimulus onset and 22 edges after the stimulus onset)

Appendix B

Matlab Code

I have divided the code for this project into two parts and put them into two folders–the main folder and the function folder. They can be found at:

https://github.com/fyz92/thesis.git

The ”main” folder contains the main programs for this project which includes all the inference steps. i.e., data simulation procedure, network inference procedure and results plot procedure. The ”function” folder contains all the functions that are called in the ”main” programs, this includes the functions to simulate the data, to resample the data, to correct for multiple comparisons etc.

However, I didn't include the original EEG data in the appendix. I don't have the permission to do that. However, it's very easy to simulate the EEG data by using the functions attached. You can play with it and generate data with the network structure you want.