Towards a new approach to reveal dynamical organization of the brain using topological data analysis

Jay Patel

The Ohio State University

patel.3316@osu.edu

April 4, 2022

Premise

- Little is known about how the brain adapts in order to efficiently complete one task versus another

Premise

- Little is known about how the brain adapts in order to efficiently complete one task versus another
- Most previous work focuses on analyzing co-fluctuations of the brain at different times[3]

Premise

- Little is known about how the brain adapts in order to efficiently complete one task versus another
- Most previous work focuses on analyzing co-fluctuations of the brain at different times[3]
- Collapsing the data into co-fluctuations will make us lose useful information about the overall dynamical organization

Premise

- Little is known about how the brain adapts in order to efficiently complete one task versus another
- Most previous work focuses on analyzing co-fluctuations of the brain at different times[3]
- Collapsing the data into co-fluctuations will make us lose useful information about the overall dynamical organization
- This paper takes the existing data and analyze it at the "single participant level"

Premise

- Little is known about how the brain adapts in order to efficiently complete one task versus another
- Most previous work focuses on analyzing co-fluctuations of the brain at different times[3]
- Collapsing the data into co-fluctuations will make us lose useful information about the overall dynamical organization
- This paper takes the existing data and analyze it at the "single participant level"
- The approach of this paper is able to probe within and between task transitions of about ($\sim$4-9 seconds)

Premise

- Little is known about how the brain adapts in order to efficiently complete one task versus another
- Most previous work focuses on analyzing co-fluctuations of the brain at different times[3]
- Collapsing the data into co-fluctuations will make us lose useful information about the overall dynamical organization
- This paper takes the existing data and analyze it at the "single participant level"
- The approach of this paper is able to probe within and between task transitions of about ($\sim$4-9 seconds)
- They observe that the revealed individual differences in the dynamical organization of the subject were predictors of the task performance

Data

- They used multiple fMRI datasets which are scans of individuals over $\sim$25 minutes while doing a variety of tasks

Data

- They used multiple fMRI datasets which are scans of individuals over $\sim$25 minutes while doing a variety of tasks
- Previously, this data has typically been used in models after some kind of "averaging" procedure

Data

- They used multiple fMRI datasets which are scans of individuals over $\sim$25 minutes while doing a variety of tasks
- Previously, this data has typically been used in models after some kind of "averaging" procedure
- The previous approaches have been unable to reveal the optimal temporal and spatial scales which best probe clinically and behaviorally relevant brain dynamics

Data

- They used multiple fMRI datasets which are scans of individuals over $\sim$25 minutes while doing a variety of tasks
- Previously, this data has typically been used in models after some kind of "averaging" procedure
- The previous approaches have been unable to reveal the optimal temporal and spatial scales which best probe clinically and behaviorally relevant brain dynamics
- Additionally, they are unable to determine if the brain dynamics are best thought of as continuous or discrete or able to tell whether a particular brain activity is healthy or not

Pipeline

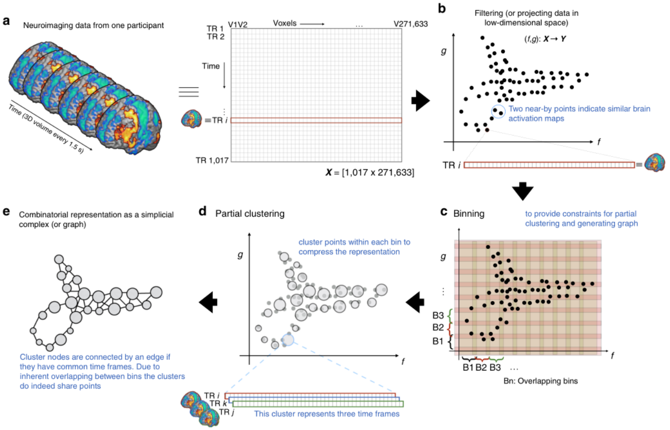


Figure: The method used to convert the 4-dimensional fMRI data into a simplicial complex. Steps b-e are a part of Mapper (the TDA-based algorithm/tool the authors used).

How Mapper works

- Step 1 the filtering into $\mathbb{R}^2$ is done using something called Stochastic Neighborhood Estimation. SNE, is used over traditional methods like PCA because SNE allows us to preserve the local structure of the original data

How Mapper works

- Step 1 the filtering into $\mathbb{R}^2$ is done using something called Stochastic Neighborhood Estimation. SNE, is used over traditional methods like PCA because SNE allows us to preserve the local structure of the original data
- Step 2 is just binning the space using the provided parameters: number of bins and the percentage overlap between bins

How Mapper works

- Step 1 the filtering into $\mathbb{R}^2$ is done using something called Stochastic Neighborhood Estimation. SNE, is used over traditional methods like PCA because SNE allows us to preserve the local structure of the original data
- Step 2 is just binning the space using the provided parameters: number of bins and the percentage overlap between bins
- Step 3 goes through each bin, and performs single-linkage clustering in order to form clusters of nearby points

How Mapper works

- Step 1 the filtering into $\mathbb{R}^2$ is done using something called Stochastic Neighborhood Estimation. SNE, is used over traditional methods like PCA because SNE allows us to preserve the local structure of the original data
- Step 2 is just binning the space using the provided parameters: number of bins and the percentage overlap between bins
- Step 3 goes through each bin, and performs single-linkage clustering in order to form clusters of nearby points
- Step 4 treats each cluster as a vertex of a graph and adds an edge between two vertices if they shared a point[4]

Filtering in Mapper

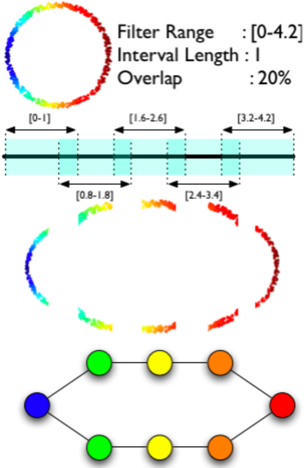


Figure: Toy example of applying a filter to data[4]. The data is sampled from a noisy circle, and the filter used is $f(x) = ||x - p||^2$, where p is the left most point in the data. We divide the range of the filter into 5 intervals which have length 1 and a 20% overlap. For each interval we compute the clustering of the points lying within the domain of the filter restricted to the interval, and connect the clusters whenever they have non-empty intersection. At the bottom is the simplicial complex which we recover whose vertices are colored by the average filter value.

Diagram of the Filtering process

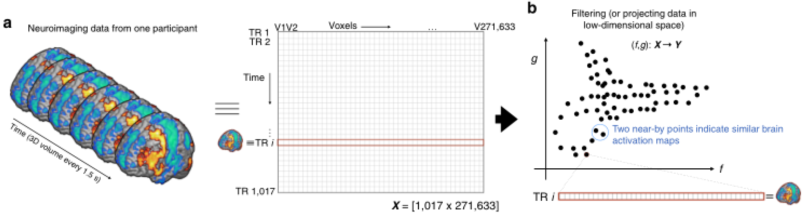


Figure: Our paper uses a very different filter function than the previous example. They use something called the Neighborhood Lens function to take their 271633-dimensional data into $\mathbb{R}^2$. This function is part of a patented software that appears to be standard.

Clustering

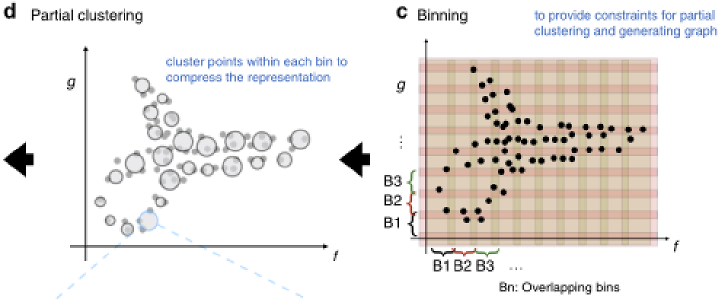


Figure: Mapper take two parameters when binning the data: Resolution and Gain. Resolution controls the number of bins and Gain controls the overlap between bins.

Example

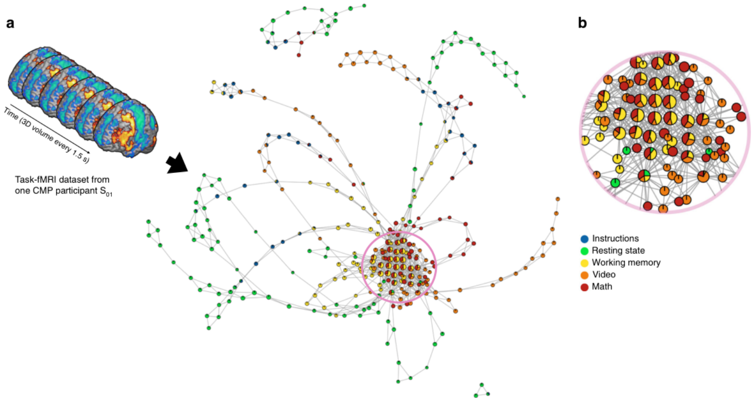


Figure: After running Mapper on an individual's fMRI data, we are left with a graph like the above. The fMRI datasets (1017 $\times$ 271633) matrix) were compressed to 279 $\pm$ 60 points.

How do we analyze these graphs?

- The most interesting and practical thing we can do with this graph is try to understand the mesoscale structure of this graph

How do we analyze these graphs?

- The most interesting and practical thing we can do with this graph is try to understand the mesoscale structure of this graph
- The most commonly analyzed mesoscale structures are community structures and core-periphery structures

How do we analyze these graphs?

- The most interesting and practical thing we can do with this graph is try to understand the mesoscale structure of this graph
- The most commonly analyzed mesoscale structures are community structures and core-periphery structures
- Note: It is necessary to have weights on the edges to perform analysis of community structure or core-periphery structure
- This paper doesn't describe how they acquire weights on the edges and there is no canonical way to assign weights from mapper

How do we analyze these graphs?

- The most interesting and practical thing we can do with this graph is try to understand the mesoscale structure of this graph
- The most commonly analyzed mesoscale structures are community structures and core-periphery structures
- Note: It is necessary to have weights on the edges to perform analysis of community structure or core-periphery structure
- This paper doesn't describe how they acquire weights on the edges and there is no canonical way to assign weights from mapper
- Two likely possibilities: the weight between two nodes is the number of timeframes they shared, or the weight is something that depends on the mean distance between the timeframes constituting the nodes

Analyzing the community structure of a graph

- You can think about the community structure as the measure of if there are many "communities" of densely connected nodes (not necessarily cliques) there are where there are only a few connections between communities

Analyzing the community structure of a graph

- You can think about the community structure as the measure of if there are many "communities" of densely connected nodes (not necessarily cliques) there are where there are only a few connections between communities
- Suppose that you have a weighted graph G and for each vertex i, we have "node strength" weight k_i on i

Analyzing the community structure of a graph

- You can think about the community structure as the measure of if there are many "communities" of densely connected nodes (not necessarily cliques) there are where there are only a few connections between communities
- Suppose that you have a weighted graph G and for each vertex i, we have "node strength" weight k_i on i
- Q_{mod} (called the modularity) is the most commonly used metric to assess the community structure of a graph[1]

Analyzing the community structure of a graph

- You can think about the community structure as the measure of if there are many "communities" of densely connected nodes (not necessarily cliques) there are where there are only a few connections between communities
- Suppose that you have a weighted graph G and for each vertex i, we have "node strength" weight k_i on i
- Q_{mod} (called the modularity) is the most commonly used metric to assess the community structure of a graph[1]

$$Q_{\mathrm{mod}} = \sum_{i,j} (A_{ij} - P_{ij})\,\delta(g_i, g_j)$$

where A is the adjacency matrix, $P_{ij} = \frac{k_i k_j}{\sum_{ij} A_{ij}}$, k_i is degree of i, g_i is the community that i belongs to, and δ is the Kronecker delta.

Some intuition for Q_{mod}

- It is based on the idea that a random graph is not expected to have a cluster structure, so the possible existence of clusters is revealed by the comparison between the actual density of edges in a subgraph and the density one would expect to have in the subgraph if the vertices of the graph were attached regardless of community structure
- This expected edge density ($P_{ij} = \frac{k_i k_j}{2m}$ where $m = \sum_{ij} A_{ij}$) depends on the chosen null model, i.e. a copy of the original graph keeping some of its structural properties but without community structure[1]

Some intuition for Q_{mod}

- It is based on the idea that a random graph is not expected to have a cluster structure, so the possible existence of clusters is revealed by the comparison between the actual density of edges in a subgraph and the density one would expect to have in the subgraph if the vertices of the graph were attached regardless of community structure
- This expected edge density ($P_{ij} = \frac{k_i k_j}{2m}$ where $m = \sum_{ij} A_{ij}$) depends on the chosen null model, i.e. a copy of the original graph keeping some of its structural properties but without community structure[1]
- In this case, the null model that produces the above expected edge density is called the configuration model. This a very standard type of model in network science because, as opposed to the Erdős-Renyi model, this model allows us to give the network arbitrary degree distributions. (The Erdős-Renyi model only allows for Poisson distribution for the degree sequence).

How Q_{mod} scales

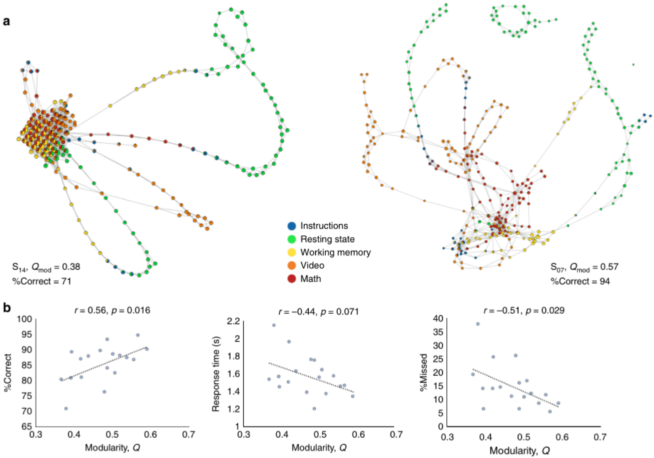


Figure: There are two different participants' shape graphs and the relations between modularity and various metrics

Analyzing the core-periphery structure of the graph

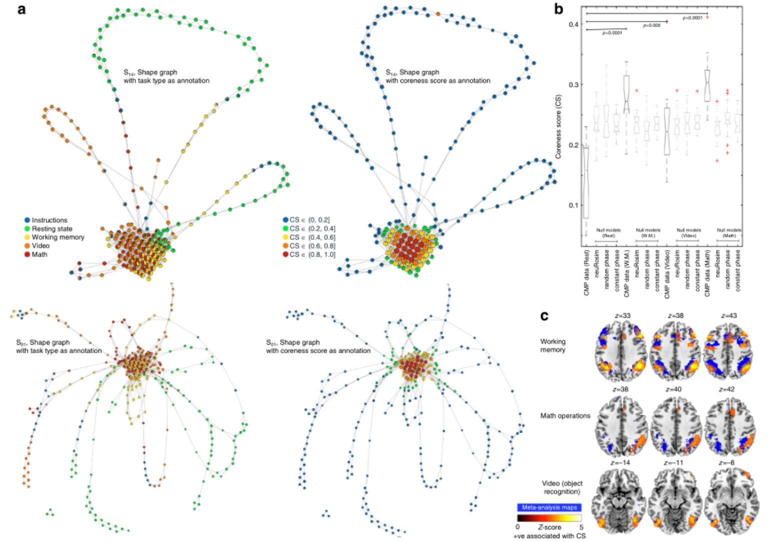
- We assign each node a coreness score (CS) by giving higher scores to nodes which lie deeper in the network
- The Borgatti-Everett algorithm is an algorithm that assigns coreness scores to each vertex of the shape graph. If we call C_i is the coreness of vertex i, then we can define the core matrix C by $C_{ij} = C_i C_j$. We can define a quality function

$$R_{(\alpha,\beta)} = \sum_{i,j} A_{ij} C_{ij}$$

where (α, β) are the two parameters which determines the boundary between the core and periphery and size of the core respectively. A large α indicates a sharp transition.
-
$$CS(i) = Z \sum_{(\alpha,\beta)} C_i(\alpha, \beta) \times R(\alpha, \beta)$$

where Z is a normalization factor to make the maximum CS be 1 and this sum is not actually over all α, β but instead over a uniform

How coreness score (CS) scales



Trying to explain the topological features using anatomy

-

Connectivity in the shape graph for different times

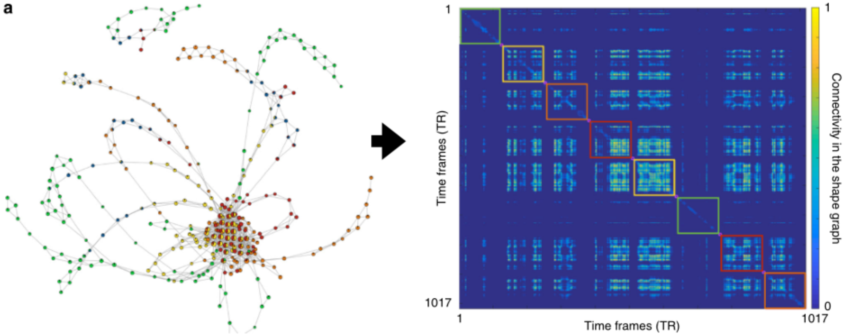


Figure: Shows which times during the scan were similar to other times by looking at the connections between times in the shape graph.

Brains are typically more connected during strenuous tasks

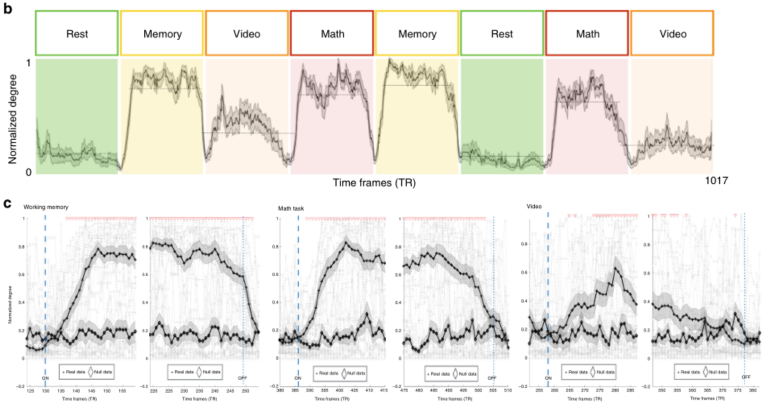


Figure: When there is a transition between tasks, this is captured by a dramatic change in the degree of the corresponding nodes. This phenomenon of nodes taken during strenuous activity having high degree is significant

References I

[1] Santo Fortunato. “Community detection in graphs”. In: *Physics Reports* 486.3-5 (2010), pp. 75–174. DOI: 10.1016/j.physrep.2009.11.002. URL: https://doi.org/10.1016%2Fj.physrep.2009.11.002.

[2] Saggar Manish et al. *Towards a new approach to reveal dynamical organization of the brain using topological data analysis*. 2018. URL: https://www.nature.com/articles/s41467-018-03664-4#citeas.

[3] Maria Giulia Preti, Thomas AW Bolton, and Dimitri Van De Ville. “The dynamic functional connectome: State-of-the-art and perspectives”. In: *NeuroImage* 160 (2017). Functional Architecture of the Brain, pp. 41–54. ISSN: 1053-8119. DOI: https://doi.org/10.1016/j.neuroimage.2016.12.061. URL: https://www.sciencedirect.com/science/article/pii/S1053811916307881.

References II

[4] Gurjeet Kaur Chatar Singh, Facundo Mémoli, and Gunnar E. Carlsson. "Topological Methods for the Analysis of High Dimensional Data Sets and 3D Object Recognition". In: (2007).

The End