

# Reviews: Topological Distances and Losses for Brain Networks

Moo K. Chung<sup>a,b,\*</sup>, Alexander Smith<sup>c</sup>, Gary Shiu<sup>d</sup>

<sup>a</sup>Department of Biostatistics and Medical Informatics, University of Wisconsin, Madison, WI, USA

<sup>b</sup>Waisman Laboratory for Brain Imaging and Behavior, University of Wisconsin, Madison, WI, USA

<sup>c</sup>Department of Chemical and Biological Engineering, University of Wisconsin, Madison, WI, USA

<sup>d</sup>Department of Physics, University of Wisconsin, Madison, WI, USA

## Abstract

Almost all statistical and machine learning methods in analyzing brain networks rely on distances and loss functions, which are mostly Euclidean or matrix norms. The Euclidean or matrix distances may fail to capture underlying subtle topological differences in brain networks. Further, Euclidean distances are sensitive to outliers. A few extreme edge weights may severely affect the distance. Thus it is necessary to use distances and loss functions that recognize topology of data. In this review paper, we survey various topological distance and loss functions from topological data analysis (TDA) and persistent homology that can be used in brain network analysis more effectively. Although there are many recent brain imaging studies that are based on TDA methods, possibly due to the lack of method awareness, TDA has not taken as the mainstream tool in brain imaging field yet. The main purpose of this paper is provide the relevant technical survey of these powerful tools that are immediately applicable to brain network data.

**Keywords:** Topological distances, topological losses, topological data analysis, persistent homology, brain networks

## 1. Introduction

There are many similarity measures, distances and loss functions used in discriminating brain networks (Banks and Carley, 1994; Chung et al., 2019c; Lee et al., 2012; Chen et al., 2016). Due to ever increasing popularity of deep learning, which estimates the parameters by minimizing loss functions, there is also a renewed interest in studying various loss functions. Many of existing distances and losses simply ignore the underlying topology of the networks and often use Euclidean distances. Existing distances often fail to capture underlying topological differences. They may lose sensitivity over topological structures such as the connected components, modules and cycles in networks. Further, Euclidean norms are sensitive to outliers. A few extreme edge weights may severely affect the distance.

In standard graph theory based brain network analysis, the similarity of networks are measured by the difference in graph theory features such as assortativity, betweenness centrality, small-worldness and network homogeneity (Bullmore and Sporns, 2009; Rubinov and Sporns, 2010; Uddin et al., 2008). Comparison of graph theory features appears to reveal changes of structural or functional connectivity associated with different clinical populations (Rubinov and Sporns, 2010). Since dense weighted brain networks are difficult to interpret and analyze, they are often turned into binary networks by thresholding edge weights (He et al., 2008; Wijk et al., 2010). The choice of thresholding the edge weights may alter the network topology. The

multiple comparison correction over every possible connections were used in determining thresholds (Rubinov et al., 2009; Salvador et al., 2005; Wijk et al., 2010). The amount of sparsity of connections were also used in determining thresholds (Bassett, 2006; He et al., 2008; Wijk et al., 2010; Lee et al., 2011c). To address the inherent problem of thresholding, persistent homological brain network analysis was developed in 2011 (Lee et al., 2011a,b). Since then the method has been successfully applied to numerous brain network studies and shown to be a powerful alternative brain network approaches (Chung et al., 2013, 2015a; Lee et al., 2012).

Persistent homology provides a coherent mathematical framework for quantifying brain images and networks (Carlsson and Memoli, 2008; Edelsbrunner and Harer, 2010b). Instead of looking at network at a fixed thresholding or resolution, persistent homology quantifies the over all changes of network topology over multiple scales (Edelsbrunner and Harer, 2010b; Horak et al., 2009; Zomorodian and Carlsson, 2005). In doing so, it reveals the most *persistent* topological features that are robust to noise and scale. Persistent homology has been applied to wide variety of data including sensor networks (Carlsson and Memoli, 2008), protein structures (Gameiro et al., 2015) and RNA viruses (Chan et al., 2013), image segmentation.

In persistent homology based brain network analysis, instead of analyzing networks at one fixed threshold that may not be optimal, we build the collection of nested networks over every possible threshold using the *graph filtration*, a persistent homological construct (Lee et al., 2011a, 2012; Chung et al., 2013, 2015a). The graph filtration is a threshold-free framework for analyzing a family of graphs but requires hierarchically building specific nested subgraph structures. The graph

\*Correspondence to: M.K.Chung, Department of Biostatistics and Medical Informatics, University of Wisconsin-Madison, 1300 University Ave, 4725 Medical Science Center, Madison, 53706, USA

Email address: mkchung@wisc.edu (Moo K. Chung)

filtration shares similarities to the existing multi-thresholding or multi-resolution network models that use many different arbitrary thresholds or scales (Achard et al., 2006; He et al., 2008; Lee et al., 2012; Kim et al., 2015; Supekar et al., 2008). Such approaches are mainly used to visually display the dynamic pattern of how graph theoretic features change over different thresholds and the pattern of change is rarely quantified. Persistent homology can be used to quantify such dynamic pattern in a more coherent mathematical framework. In numerous studies, persistent homological network approach is shown to be very robust and outperforming many existing network measures and methods. In Lee et al. (2011a, 2012), persistent homology was shown to outperform against eight existing graph theory features such as assortativity, between centrality, clustering coefficient, characteristic path length, small-worldness, modularity and global network homogeneity. In Chung et al. (2017a), persistent homology was shown to outperform various matrix norms. In Wang et al. (2018), persistent homology using the persistent landscape was shown to outperform power spectral density and local variance methods. In Wang et al. (2017b), persistent homology was shown to outperform topographic power maps. In Yoo et al. (2017), center persistency was shown to outperform the network-based statistic and element-wise multiple corrections.

Existing statistical and machine learning methods for brain networks are based on distance and loss functions that are mostly Euclidean based or matrix norms. Such distances are geometric in nature and not sensitive enough for topological signal differences often observed in brain networks. Thus, it is necessary to use distances and loss functions that are topologically more sensitive. Persistent homology offers a coherent mathematical framework for measuring network distances *topologically*. Various topological distances such as Gromov-Hausdorff (GH) distance (Tuzhilin, 2016; Carlsson and Memoli, 2008; Carlsson and Memoli, 2010; Chazal et al., 2009; Lee et al., 2011a, 2012), bottleneck distances (Lee et al., 2012, 2017; Chung et al., 2015a) and the Kolmogorov-Smirnov (KS) distance (Chung, 2012; Chung et al., 2017b; Lee et al., 2017) are available. The main purpose of this paper is to review such topological distances within the context of persistent homology.

## 2. Traditional distances and losses

### 2.1. Losses in statistics and machine learning

In statistical analysis and machine learning, the loss function acts as a cost function, which needs to be minimized to determine the model fit. Widely used loss functions are often Euclidean, and have proven effective in traditional applications (Bishop, 2006). Here, we describe the most often used loss functions that were applied in various brain image analysis tasks.

The mean square error loss (MSE) is the most often used loss. For a given model with  $n$  input data  $x_i$ , input labels  $y_i$ , and model outputs  $\hat{y}_i$ , MSE is given as

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (1)$$

The MSE is based upon the  $L_2$ -norm between the model prediction and input data and the most often used loss in regression. In regression setting, MSE was used in identifying brain imaging predictions for memory performance (Wang et al., 2011), estimation of kinetic constants from PET data (O'Sullivan and Saha, 1999), for correction partial volume effects in arterial spin labeling MRI (Kim et al., 2018; Asllani et al., 2008), EEG signal classification with neural networks (Kottaimalai et al., 2013). This is also the basis of widely used  $k$ -means clustering (Jain et al., 1999) in identifying abnormalities in brain tumors (Arunkumar et al., 2019), modeling state spaces in rsfMRI (Huang et al., 2020b). This loss was also used for image classification of autism spectrum disorder (Heinsfeld et al., 2018), tumor segmentation for MR brain images (Mittal et al., 2019) among other learning tasks.

One problem associated with MSE is that it heavily penalizes outliers due to its quadratic loss (Bishop, 2006). The impact of outliers can be significantly reduced using the mean absolute error (MAE), which is  $L_1$ -norm:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (2)$$

The MAE was successfully used in clustering (Jain et al., 1999), tumor segmentation in MRI (Blessy and Sulochana, 2015), age prediction in deep learning (Cole et al., 2017) and the conversion of MRI to CT through deep learning (Wolterink et al., 2017).

Another common loss often used in probabilistic model building is the cross entropy  $L_{ent}$ , which was initially used in logistic regression and artificial neural networks. The output of the model is defined as a probability  $p_i$  that a given input  $x_i$  belongs to a binary class:

$$L_{ent} = y_i \log p_i + (1 - y_i) \log (1 - p_i). \quad (3)$$

The cross entropy is equivalent to the maximum likelihood of the product of Bernoulli distributions. The use of cross entropy over MSE can lead to faster training times and improved model generalization (Bishop, 2006). The cross entropy has been used in improving brain MRI segmentation (Moeskops et al., 2017), the prediction of intracranial pressure levels after brain injury (Chen et al., 2010) and in a machine learning interface for medical image analysis (Zhang and Kagen, 2017). Logistic regression has found use in a deep learning ensemble regression model for brain disease diagnosis (Suk et al., 2017) and in the classification of brain hemorrhages with the convolutional neural networks (Jnawali et al., 2018).

Another broadly applied loss in the support vector machine and other classification tasks is the *hinge loss*, which uses the  $L_\infty$ -norm (Bishop, 2006). The hinge loss heavily penalizes incorrect classifications or classifications that are correct but are near the decision boundary. The hinge loss has been used in the classification of CT brain images with deep learning networks for identification of Alzheimer's disease (Gao et al., 2017). It has also been applied to data collected from working brains to guide machine learning algorithms (Fong et al., 2018).

## 2.2. Matrix norms as network distances

Many distance or similarity measures are not metrics but having metric distances makes networks more stable due to the triangle inequality. Further, existing network distance concepts are often borrowed from the metric space theory. Let us start with formulating brain networks as metric spaces. The brain networks are often algebraically represented as matrices of size  $p \times p$ , where  $p$  is predetermined number of parcellations. Often 100-300 parcellations are used for this purpose. It is necessary to use distances or losses defined on the connectivity matrices. Consider a weighted graph or network  $\mathcal{X} = (V, w)$  with the node set  $V = \{1, \dots, p\}$  and the edge weights  $w = (w_{ij})$ , where  $w_{ij}$  is the weight between nodes  $i$  and  $j$ . The edge weight is usually given by a similarity measure between the observed data on the nodes. Various similarity measures have been proposed. The correlation or mutual information between measurements for the biological or metabolic network and the frequency of contact between actors for the social network have been used as edge weights (Bassett, 2006; Bien and Tibshirani, 2011). We may assume that the edge weights satisfy the metric properties: nonnegativity, identity, symmetry and the triangle inequality such that

$$w_{i,j} \geq 0, w_{ii} = 0, w_{ij} = w_{ji}, w_{ij} \leq w_{ik} + w_{kj}.$$

With these conditions,  $\mathcal{X} = (V, w)$  forms a metric space. Although the metric property is not necessary for building a network, it offers many nice mathematical properties and easier interpretation on network connectivity. Further, persistent homology is often built on top of metric spaces. Many real-world networks satisfy the metric properties.

Given measurement vector  $\mathbf{x}_i = (x_{1i}, \dots, x_{ni})^T \in \mathbb{R}^n$  on the node  $i$ . The weight  $w = (w_{ij})$  between nodes is often given by some bivariate function  $f$ :  $w_{ij} = f(\mathbf{x}_i, \mathbf{x}_j)$ . The correlation between  $\mathbf{x}_i$  and  $\mathbf{x}_j$ , denoted as  $\text{corr}(\mathbf{x}_i, \mathbf{x}_j)$ , is a bivariate function. If the weights  $w = (w_{ij})$  are given by

$$w_{ij} = \sqrt{1 - \text{corr}(\mathbf{x}_i, \mathbf{x}_j)},$$

it can be shown that  $\mathcal{X} = (V, w)$  forms a metric space.

Matrix norm of the difference between networks is often used as a measure of similarity between networks (Banks and Carley, 1994; Zhu et al., 2014). Given two networks  $\mathcal{X}^1 = (V, w^1)$  and  $\mathcal{X}^2 = (V, w^2)$ , the  $L_l$ -norm of network difference is given by

$$D_l(\mathcal{X}^1, \mathcal{X}^2) = \|w^1 - w^2\|_l = \left( \sum_{i,j} |w_{ij}^1 - w_{ij}^2|^l \right)^{1/l}.$$

Note  $L_l$  is the element-wise Euclidean distance in  $l$ -dimension. When  $l = \infty$ ,  $L_\infty$ -distance is written as

$$D_\infty(\mathcal{X}^1, \mathcal{X}^2) = \|w^1 - w^2\|_\infty = \max_{i,j} |w_{ij}^1 - w_{ij}^2|.$$

The element-wise differences may not capture additional higher order similarity. For instance, there might be relations between a pair of columns or rows (Zhu et al., 2014). Also  $L_1$  and  $L_2$ -distances usually suffer the problem of outliers. Few outlying

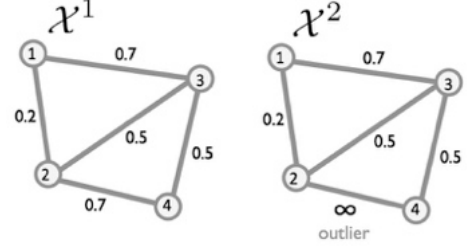


Figure 1: Toy networks with an outlying edge weight. All the  $L_l$ -norm based distance will give  $\infty$  distance while topological distances may not be severely affected with such outlying data.

extreme edge weights may severely affect the distance. Given two identical networks that only differ in one outlying edge with infinite weight (Figure 1),  $L_l(\mathcal{X}^1, \mathcal{X}^2) = \infty$  and  $L_\infty(\mathcal{X}^1, \mathcal{X}^2) = \infty$ . Thus, the usual matrix norm based distance is sensitive to even a single outlying edge. However, topological distances are not sensitive to edges weights but sensitive to the underlying topology. Further, these distances ignore the underlying topological structures. Thus, there is a need to define distances that are more topological.

## 2.3. Log-Euclidean distance

The log-Euclidean was previously used in measuring distance between correlation based brain networks (Qiu et al., 2015; Chung et al., 2015a), which are supposed to reside in the space of positive definite symmetric (PDS) matrices. Let  $\mathcal{S}_p$  be the space of symmetric (PDS) matrices of size  $p \times p$ . Let  $\mathcal{P}_p$  be the space of positive definite symmetric (PDS) matrices. Note that the dimension of  $\mathcal{S}_p$  and  $\mathcal{P}_p$  is  $p(p+1)/2$ , i.e.,  $\mathcal{S}_p, \mathcal{P}_p \subset \mathbb{R}^{p(p+1)/2}$ . While  $\mathcal{S}_p$  is a flat Euclidean space,  $\mathcal{P}_p$  is a curved manifold. Even though, various computation in  $\mathcal{P}_p$  is fairly involving, the use of exponential map  $\exp : \mathcal{S}_p \rightarrow \mathcal{P}_p$  and its inverse make computations in  $\mathcal{P}_p$  tractable. The exponential map is realized with matrix exponential. In practice, the singular value decomposition is mainly used in computing the matrix exponential as follows. For  $X \in \mathcal{S}_p$ , there exists an orthogonal matrix  $Q$  such that

$$X = Q^T \Lambda Q$$

with  $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_p)$ , the diagonal matrix consisting of entries  $\lambda_1, \dots, \lambda_p$ . Then matrix exponential is defined as

$$\exp X = Q^T (\exp \Lambda) Q.$$

Thus  $\exp X = Q^T (\exp \Lambda) Q$ , where  $\exp \Lambda$  is the diagonal matrix consisting of entries  $e^{\lambda_1}, \dots, e^{\lambda_p}$ . Similarly, for  $Y \in \mathcal{P}_m$ , we have decomposition  $Y = Q^T \Lambda Q$  with  $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_p)$ . Then the matrix logarithm of  $Y$  is computed as

$$\log Y = Q^T (\log \Lambda) Q.$$

If matrix  $Y$  is nonnegative definite with zero eigenvalues, the matrix logarithm is *not* defined since  $\log 0$  is not defined. Thus, we cannot apply logarithm directly to rank-deficient large correlation and covariance matrices obtained from small number of subjects. One way of applying logarithm to nonnegative

definite matrices is to make matrix  $Y$  diagonally dominant by adding a diagonal matrix  $\alpha I$  with suitable choice of relatively large  $\alpha$  (Chan and Wood, 1997). Alternately, we can perform a graphical LASSO-type of sparse model and obtain the closest possible positive definite matrices (Chung et al., 2015a; Qiu et al., 2015; Mazumder and Hastie, 2012).

For  $X, Y \in \mathcal{P}_p$ , the log-Euclidean metric is given by the Frobenius norm (Qiu et al., 2015)

$$D_{LE}(X, Y) = \|\log X - \log Y\|_F = \sqrt{\text{tr}(\log(X) - \log(Y))^2}.$$

The log-Euclidean distance was used in Qiu et al. (2015) to compute the mean of functional brain networks and the variation of each individual network around the mean. If  $C_i$  denotes the brain network represented as the PDS correlation matrix of the  $i$ -th subject, the average of all brain network within log-Euclidean framework is given by

$$\bar{C} = \exp\left(\frac{1}{n} \sum_{i=1}^n \log C_i\right).$$

The metric was also used in the local linear embedding of brain functional networks (Qiu et al., 2015) and regression for resting-state functional brain networks in the PDS space (Huang et al., 2020b).

#### 2.4. Graph matching

Graph matching is well formulated established method for matching two different graphs via combinatorial optimization. The method is often used in distributed controls and computer vision (Zavlanos and Pappas, 2008). Suppose two graphs  $X_1 = (V, w_1)$  and  $X_2 = (V, w_2)$  with  $p$  nodes are given. If  $A_1$  and  $A_2$  are adjacency matrices of  $X_1$  and  $X_2$ , the graph matching cost function is given by

$$D_{GM}(X_1, X_2) = \min_Q \|QA_1Q^\top - A_2\|_2^2,$$

where  $Q$  is the permutation matrix that shuffles the node index. Two graphs are *isomorphic* if  $D_{GM} = 0$ . All isomorphic graphs have the same structure, since one obtain  $A_2$  from  $A_1$  by relabelling of the nodes. However, not every two graphs are isomorphic. If  $\lambda_1^i \geq \lambda_2^i \geq \dots \geq \lambda_p^i$  are eigenvalues of  $A_i$ . Then we can show that (Zavlanos and Pappas, 2008)

$$\|QA_1Q^\top - A_2\|_2^2 \geq \sum_{i=1}^p (\lambda_1^i - \lambda_2^i)^2,$$

which is the lower bound for the graph matching cost  $D_{GM}$ .

The main limitation of graph matching is the exponential run time  $2^{O(\sqrt{p \log p})}$ , which does not scale well for large  $p$  (Babai and Luks, 1983). The other limitation is that if the size of node sets don't match, it is difficult to apply the graph matching algorithm. Additional nodes are argumented to match the node sets but the argumentation can be somewhat arbitrary (Guo and Srivastava, 2020). The graph matching has been used in matching and averaging heterogenous tree structures such as brain artery trees and neuronal trees (Guo and Srivastava, 2020). If

the same brain parcellation is used across subjects, we do not need realignment of node labels so graph matching has not seen many applications in brain network analysis. However, with the availability of many different parcellations, it might be useful matching networks across different parcellations.

#### 2.5. Canonical correlations

Many existing approaches for measuring distance between brain networks assume the size of networks to be the same. Otherwise, it is difficult to define distance and loss functions. The canonical correlation is perhaps one of few statistical similarly measure that enable to compute the distance between the measurements of different sizes (Hotelling, 1992). It might be useful for comparing brain networks obtained parcellations of different sizes.

Given two vectors  $X = (x_1, \dots, x_n)^\top$  and  $Y = (y_1, \dots, y_m)^\top$ , the canonical correlation between  $X$  and  $Y$  is given by

$$\rho = \max_{a,b} \text{corr}(a^\top X, b^\top Y).$$

Various numerical methods are available for computing  $\rho$ . It is computed using `canocorr` in MATLAB and `cancor` or `CCA` in R package. For brain connectivity matrices, we vectorize only the upper triangle components of the connectivity matrices and compute the canonical correlations. Numerically, the canonical correlation is mainly computed vis the singular value decomposition (SVD).

The canonical correlation has been widely used in various applications including deep learning (Andrew et al., 2013), multiview clustering (Chaudhuri et al., 2009). In brain imaging, it is mainly used in correlating measurements from different modalities. In Avants et al. (2010), fractional anisotropy values from diffusion tensor imaging and cortical thickness from MRI were correlated using canonical correlations. In Correa et al. (2009), fMRI, structural MRI and EEG measurements are correlated using canonical correlations.

### 3. Preliminary: Persistent homology

We start with the basic mathematical understanding of persistent homology.

#### 3.1. Simplicial homology

A high dimensional object can be approximated by the point cloud data  $X$  consisting of  $p$  number of points. If we connect points of which distance satisfy a given criterion, the connected points start to recover the topology of the object. Hence, we can represent the underlying topology as a collection of the subsets of  $X$  that consists of nodes which are connected (Edelsbrunner and Harer, 2010b; Hart, 1999). Given a point cloud data set  $X$  with a rule for connections, the topological space is a simplicial complex and its element is a simplex (Zomorodian, 2009). For point cloud data, the Delaunay triangulation is probably the most widely used method for connecting points. The Delaunay triangulation represents the collection of points in space as



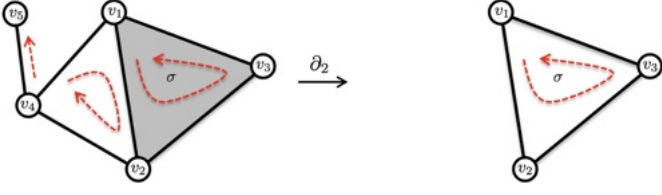


Figure 2: A simplicial complex with 5 vertices and 2-simplex  $\sigma = [v_1, v_2, v_3]$  with a filled-in face (colored gray). After boundary operation  $\partial_2$ , we are only left with 1-simplices  $[v_1, v_2] + [v_2, v_3] + [v_3, v_1]$ , which is the boundary of the filled in triangle. The complex has a single connected component ( $\beta_0 = 1$ ) and a single 1-cycle. The dotted red arrows are the orientation of simplices.

a graph whose face consists of triangles. Another way of connecting point cloud data is based on Rips complex often studied in persistent homology.

Homology is an algebraic formalism to associate a sequence of objects with a topological space (Edelsbrunner and Harer, 2010b). In persistent homology, the algebraic formalism is usually built on top of objects that are hierarchically nested such as morse filtration, graph filtration and dendrograms. Formally homology usually refers to homology groups which are often built on top of a simplicial complex for point cloud and network data (Lee et al., 2014).

The  $k$ -simplex  $\sigma$  is the convex hull of  $v + 1$  independent points  $v_0, \dots, v_k$ . A point is a 0-simplex, an edge is a 1-simplex, and a filled-in triangle is a 2-simplex. A *simplicial complex* is a finite collection of simplices such as points (0-simplex), lines (1-simplex), triangles (2-simplex) and higher dimensional counter parts (Edelsbrunner and Harer, 2010b). A  $k$ -skeleton is a simplex complex of up to  $k$  simplices. Hence a graph is a 1-skeleton consisting of 0-simplices (nodes) and 1-simplices (edges). There are various simplicial complexes. The most often used simplicial complex in persistent homology is the Rips complex.

The boundary operations have been very useful for effectively quantifying persistent homology. Let  $C_k$  be the collection of  $k$ -simplices. We define the  $k$ -th boundary operator

$$\partial_k : C_k \rightarrow C_{k-1}.$$

that removes the filled-in interior of  $k$ -simplices. Consider a filled-in triangle  $\sigma = [v_1, v_2, v_3] \in C_2$  with three vertices  $v_1, v_2, v_3$  in Figure 2. The boundary operator  $\partial_k$  applied to  $\sigma$  resulted in the collection of three edges that forms the boundary of  $\sigma$ :

$$\partial_2 \sigma = [v_1, v_2] + [v_2, v_3] + [v_3, v_1] \in C_1. \quad (4)$$

If we give the direction or orientation to edges such that

$$[v_3, v_1] = -[v_1, v_3],$$

and use edge notation  $e_{ij} = [v_i, v_j]$ , we can write (4) as

$$\partial_2 \sigma = e_{12} + e_{23} + e_{31} = e_{12} + e_{23} - e_{13}.$$

We can apply the boundary operation  $\partial_1$  further to  $\partial_2 \sigma$  and obtain

$$\begin{aligned} \partial_1 \partial_2 \sigma &= \partial_1 e_{12} + \partial_1 e_{23} + \partial_1 e_{31} \\ &= v_2 - v_1 + v_3 - v_2 + v_1 - v_3 = 0. \end{aligned}$$

The boundary operation twice will results in an empty set. Such algebraic representation for boundary operation has been very useful for effectively quantifying persistent homology.

The boundary operator  $\partial_k$  can be represented using the *boundary matrix*, which is the higher dimensional version of the incidence matrix. (Lee et al., 2018, 2019; Schaub et al., 2018) showing how  $(k-1)$ -dimensional simplices are forming  $k$ -dimensional simplex. The boundary matrix  $\partial_k$  is a matrix of size  $n \times m$  consisting of ones and zeros, where  $n$  is the total number of  $(k-1)$ -dimensional simplices and  $m$  is the total number of  $k$ -dimensional simplices in the simplicial complex. Index the  $(k-1)$ -dimensional simplices as  $\sigma_1, \dots, \sigma_n$  and  $k$ -dimensional simplices as  $\tau_1, \dots, \tau_m$ . The  $(i, j)$ -th entry of  $\partial_k$  is one if  $\sigma_i$  is a face of  $\tau_j$  otherwise zero. The entry can be -1 depending on the orientation of  $\sigma_i$ .

For the simplicial complex in Figure 2 the boundary matrices are given by

$$\begin{aligned} \partial_2 &= \begin{matrix} & \sigma \\ e_{12} & \begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \\ e_{23} & \\ e_{31} & \\ e_{24} & \\ e_{41} & \\ e_{45} & \end{matrix} \\ \partial_1 &= \begin{matrix} & v_1 & v_2 & v_3 & v_4 & v_5 \\ e_{12} & \begin{pmatrix} -1 & 0 & 1 & 0 & 1 & 0 \\ 1 & -1 & 0 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \\ e_{23} & \\ e_{31} & \\ e_{24} & \\ e_{41} & \\ e_{45} & \end{matrix} \\ \partial_0 &= \begin{matrix} & v_1 & v_2 & v_3 & v_4 & v_5 \\ 0 & \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{matrix} \end{aligned}$$

Although we put direction in the boundary matrix  $\partial_1$  by adding sign, the Betti number  $\beta_1$  computation will be invariant. With boundary operations, we can build a vector space  $C_k$  using the set of  $k$ -simplices as a basis. The vector spaces  $C_k, C_{k-1}, C_{k-2}, \dots$  are then sequentially nested by boundary operator  $\partial_k$  (Edelsbrunner and Harer, 2010b):

$$\dots \xrightarrow{\partial_{k+1}} C_k \xrightarrow{\partial_k} C_{k-1} \xrightarrow{\partial_{k-1}} C_{k-2} \xrightarrow{\partial_{k-2}} \dots \quad (5)$$

Such nested structure is called the *chain complex*. Let  $B_k$  be a collection of boundaries obtained as the image of  $\partial_k$ , i.e.,

$$B_k = \{\partial_k \sigma : \sigma \in C_k\}.$$

Let  $Z_k$  be a collection of cycles obtained as the kernel of  $\partial_k$ , i.e.,

$$Z_k = \{\sigma \in C_k : \partial_k \sigma = 0\}.$$

For instance, the 1-cycle formed by edges  $e_{12}, e_{23}, e_{31}$  in Figure 2 is the boundary of the filled-in gray colored triangle  $\sigma$ . The boundaries  $B_k$  form subgroups of the cycles  $Z_k$ , i.e,  $B_k \subset Z_k$ . We can partition  $Z_k$  into cycles that differ from each other by boundaries through the quotient space

$$H_k = Z_k / B_k,$$

which is called the  $k$ -th homology group. The elements of the  $k$ -th homology group are often referred to as  $k$ -dimensional cycles or  $k$ -cycles. The  $k$ -th Betti number  $\beta_k$  is then the number of  $k$ -dimensional cycles, which is given by the rank of  $H_k$ , i.e.,

$$\beta_k = \text{rank}(H_k) = \text{rank}(Z_k) - \text{rank}(B_k). \quad (6)$$

The 0-th Betti number is the number of connected components while 1-st Betti number is the number of cycles.

The Betti numbers  $\beta_k$  are usually algebraically computed by reducing the boundary matrix  $\partial_k$  to the Smith normal form, which has a block diagonal matrix as a submatrix in the upper left, via Gaussian elimination (Edelsbrunner and Harer, 2010b). For instance, the boundary matrices  $\partial_i$  in Figure 2 is transformed to the Smith normal form  $S(\partial_i)$  after Gaussian elimination as

$$S(\partial_1) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad S(\partial_2) = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

In the Smith normal form  $S(\partial_k)$ , the number of columns containing only zeros is  $\text{rank}(Z_k)$ , the number of  $k$ -cycles while the number of rows containing one is  $\text{rank}(B_{k-1})$ , the number of  $(k-1)$ -cycles that are boundaries. From (6), the Betti number computation involves the rank computation of two boundary matrices. In Figure 2 example, there are  $\text{rank}(Z_1) = 2$  zero columns and  $\text{rank}(B_0) = 4$  non-zero rows.  $\text{rank}(Z_0) = 5$  is trivially the number of nodes in the simplicial complex while there are  $\text{rank}(B_1) = 1$  for  $S(\partial_2)$ . Thus, we have

$$\begin{aligned} \beta_0 &= \text{rank}(Z_0) - \text{rank}(B_0) = 5 - 4 = 1, \\ \beta_1 &= \text{rank}(Z_1) - \text{rank}(B_1) = 2 - 1 = 1. \end{aligned}$$

The Betti numbers can be also computed using the Hodge Laplacian without Gaussian elimination. The standard graph Laplacian is defined as

$$\Delta_0 = \partial_1 \partial_1^\top,$$

which is also called the 0-th Hodge Laplacian (Lee et al., 2018). In general, the  $k$ -th Hodge Laplacian is defined as

$$\Delta_k = \partial_{k+1} \partial_{k+1}^\top + \partial_k \partial_k^\top.$$

The boundary operation  $\partial_k$  only depends on  $k$ -simplices. Thus,  $\Delta_k$  is uniquely determined by  $(k+1)$ - and  $k$ -simplices. The  $k$ -th Laplacian is sparse a  $n_k \times n_k$  positive semi-definite symmetric matrix, where  $n_k$  is the number of  $k$ -simplices in the network (Friedman, 1998). Then the  $k$ -th Betti number  $\beta_k$  is the dimension of  $\ker \Delta_k$ , which is given by computing the rank of  $\Delta_k$ . The 0th Betti number, the number of connected component, is computed from  $\Delta_0$  while the 1st Betti number, the number of cycles, is computed from  $\Delta_1$ .

The  $k$ -th hodge Laplacian depends only on  $n_k$  number of  $k$ -simplices in the data. After lengthy algebraic derivation, we can show that

$$\Delta_k = D - A_u + (k+1)I_{n_k} + A_l,$$

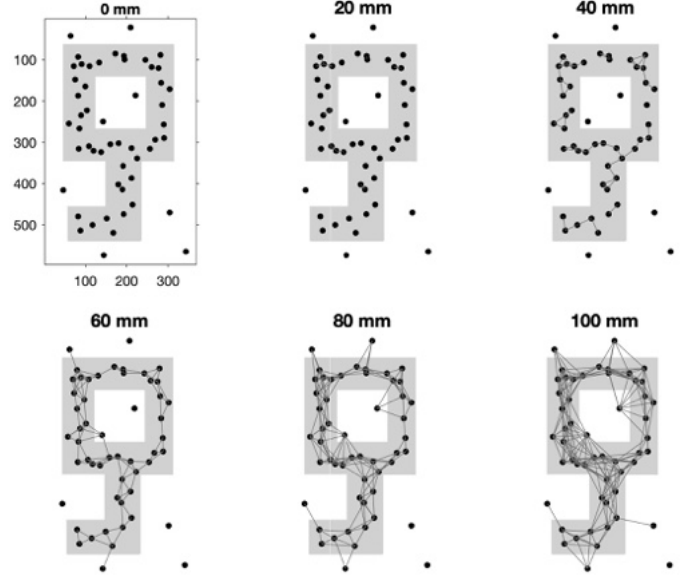


Figure 3: Rips filtration on 1-skeleton (Rips complex consisting of only nodes and edges) of the point cloud data that was sampled along the underlying key shaped data. If two points are within the given radius, we connect them with an edge.

where  $A_u$  and  $A_l$  are the upper and lower adjacency matrices between the  $k$ -simplices.  $D = \text{diag}(\deg(\sigma_1), \dots, \deg(\sigma_{n_k}))$  is the diagonal matrix consisting of the sum of node degrees of simplices  $\sigma_j$  (Muhammad and Egerstedt, 2006).

### 3.2. Rips filtrations: filtrations on point cloud data

The Rips complex has been the main building block for persistent homology and defined on top of the point cloud data (Christ, 2008). The *Rips complex* is a simplicial complex constructed by connecting two data points if they are within specific distance  $\epsilon$ . Figure 3 shows an example of the Rips complex that approximates the gray object with a point cloud. Given a point cloud data, the Rips complex  $R_\epsilon$  is a simplicial complex whose  $k$ -simplices correspond to unordered  $(k+1)$ -tuples of points which are pairwise within distance  $\epsilon$  (Christ, 2008). While a graph has at most 1-simplices, the Rips complex has at most  $k$ -simplices. The Rips complex has the property that

$$\mathcal{R}_{\epsilon_0} \subset \mathcal{R}_{\epsilon_1} \subset \mathcal{R}_{\epsilon_2} \subset \dots$$

for  $0 = \epsilon_0 \leq \epsilon_1 \leq \epsilon_2 \leq \dots$ . When  $\epsilon = 0$ , the Rips complex is simply the node set  $V$ . By increasing the filtration value  $\epsilon$ , we are connecting more nodes so the size of the edge set increases. Such the nested sequence of the Rips complexes is called a *Rips filtration*, the main object of interest in the persistent homology (Edelsbrunner and Harer, 2008). The increasing  $\epsilon$  values are called the filtration values.

One major problem of the Rips complex is that as the number of vertices  $p$  increase, the resulting simplicial complex becomes very dense. Further, as the filtration values increases, there exists an edge between every pair of vertices and filled triangle between every triple of vertices. At higher filtration val-

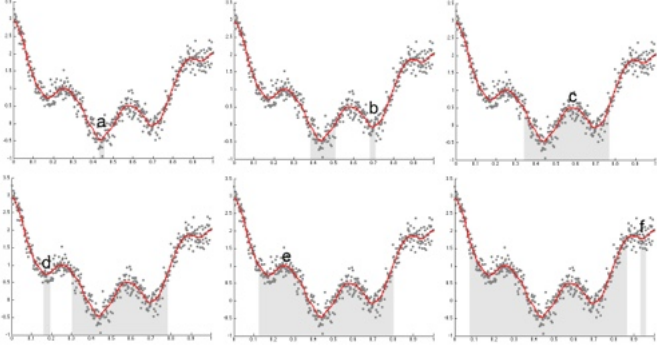


Figure 4: The births and deaths of connected components in the sublevel sets in a Morse filtration (Chung et al., 2009a). We have local minimums  $a < b < d < f$  and local maximums  $c < e$ . At  $y = a$ , we have a single connected component (gray area). As we increase the filtration value to  $y = b$ , we have the birth of a new component (second gray area). At the local maximum  $y = c$ , the two sublevel sets merge together to form a single component. This is viewed as the death of a component. The process continues till we exhaust all the critical values. Following the Elder rule, we pair birth to death:  $(c, b)$  and  $(e, d)$ . Other critical values are paired similarly. These paired points form the persistent diagram (PD).

ues, Rips filtration becomes very ineffective representation of data.

### 3.3. Morse filtrations and elder rule

A function is called a *Morse function* if all critical values are distinct and non-degenerate, i.e., the Hessian does not vanish (Milnor, 1973). For a 1D Morse function  $y = f(t)$ , define sublevel set  $R(y)$  as

$$R(y) = f^{-1}(-\infty, y] = \{t \in \mathbb{R} : f(t) < y\}.$$

The sublevel set is the domain of  $f$  satisfying  $f(t) \leq y$ . As we increase height  $y_1 \leq y_2 \leq y_3 \leq \dots$ , the sublevel set gets bigger such that

$$R(y_1) \subset R(y_2) \subset R(y_3) \subset \dots$$

The sequence of the sublevel sets form a *Morse filtration* with filtration values  $y_1, y_2, y_3, \dots$ . Let  $\beta_0(y)$  be the 0-th Betti number of  $R(y)$ .  $\beta_0(y)$  counts the number of connected components in  $R(y)$ . The number of connected components is the most often used topological invariant in applications (Edelsbrunner and Harer, 2008).  $\beta_0(y)$  only changes its value as it passes through critical values (Figure 4). The birth and death of connected components in the Morse filtration is characterized by the pairing of local minimums and maximums. For 1D Morse functions, we do not have higher dimensional topological features beyond the connected components.

Let us denote the local minimums as  $g_1, \dots, g_m$  and the local maximums as  $h_1, \dots, h_n$ . Since the critical values of a Morse function are all distinct, we can combine all minimums and maximums and reorder them from the smallest to the largest: We further order all critical values together and let

$$g_1 = z_{(1)} < z_{(2)} < \dots < z_{(m+n)} = h_n,$$

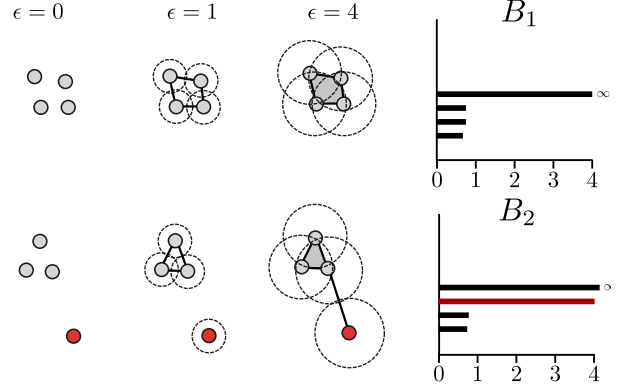


Figure 5: A comparison of the barcodes obtained from a sample with no measurement error (top) and a sample containing a single measurement error (red point). While both point clouds may be taken from the same population, the presence of a single measurement error can dramatically impact the resulting barcodes.  $B_1$  has most features with approximately equal lifetimes ( $\tau_i - \xi_i$ ), where as  $B_2$  has a feature with a large lifetime corresponding to the improperly measured point. Simply looking at the barcodes, it is difficult to differentiate topological signal to noise.

where  $z_i$  is either  $h_i$  or  $g_i$  and  $z_{(i)}$  denotes the  $i$ -th largest number in  $z_1, \dots, z_{m+n}$ . In a Morse function,  $g_1$  is smaller than  $h_1$  and  $g_m$  is smaller than  $h_n$  in the unbounded domain  $\mathbb{R}$  (Chung et al., 2009a).

By keeping track of the birth and death of components, it is possible to compute topological invariants of sublevel sets such as the 0-th Betti number  $\beta_0$  (Edelsbrunner and Harer, 2008). As we move  $y$  from  $-\infty$  to  $\infty$ , at a local minimum, the sublevel set adds a new component so that

$$\beta_0(g_i - \epsilon) = \beta_0(g_i) + 1$$

for sufficiently small  $\epsilon$ . This process is called the *birth* of the component. The newly born component is identified with the local minimum  $g_i$ .

Similarly for at a local maximum, two components are merged as one so that

$$\beta_0(h_i - \epsilon) = \beta_0(h_i) - 1.$$

This process is called the *death* of the component. Since the number of connected components will only change if we pass through critical points and we can iteratively compute  $\beta_0$  at each critical value as

$$\beta_0(z_{(i+1)}) = \beta_0(z_{(i)}) \pm 1.$$

The sign depends on if  $z_{(i)}$  is maximum ( $-1$ ) or minimum ( $+1$ ). This is the basis of the Morse theory (Milnor, 1973) that states that the topological characteristics of the sublevel set of Morse function are completely characterized by critical values.

To reduce the effect of low signal-to-noise ratio and to obtain smooth Morse function, either spatial or temporal smoothing have been often applied to brain imaging data before persistent homology is applied. In Chung et al. (2015a); Lee et al. (2017), Gaussian kernel smoothing was applied to 3D volumetric images. In Wang et al. (2018), diffusion was applied to temporally smooth data.

### 3.4. Persistent diagrams & barcodes

In persistent homology, the topology of underlying data can be represented by the birth and death of cycles. In a graph, the 0D and 1D cycles are a connected component and a cycle (Carlsson et al., 2008). During a filtration, cycles in a homology group appear and disappear. If a cycle appears at birth value  $\xi$  and disappears at death value  $\tau$ , it can be encoded into a point,  $(\xi, \tau)$  ( $0 \leq \xi \leq \tau < \infty$ ) in  $\mathbb{R}^2$ . If  $m$  number of cycles appear during the filtration of a network  $X = (V, w)$ , the homology group can be represented by a point set

$$\mathcal{P}(X) = \{(\xi_1, \tau_1), \dots, (\xi_m, \tau_m)\}.$$

This scatter plot is called the persistence diagram (PD) (Cohen-Steiner et al., 2007). A *barcode* encodes the birth time  $\xi_i$  and death time  $\tau_i$  of the cycle as an interval  $(\xi_i, \tau_i)$ . The PD and barcode encode topologically equivalent information. The length of a bar  $\tau_i - \xi_i$  is the *persistence* of the cycle (Figure 5) (Edelsbrunner and Harer, 2010b). Longer bars corresponds to more persistent cycles that are considered as topological signal while shorter bars correspond to topological noise. However, recently such interpretation has been disputed and it may possible to have meaningful signal even in shorter bars (Bubenik et al., 2020).

The birth and death of connected components in the sublevel set of a function can be quantified and visualized by persistent diagram (PD) (Figure 4). Consider a smooth Morse function  $y = f(x)$  with unique critical values  $a < b < c < d < e < f$  that estimate the underlying data. Such function can be estimated using kernel smoothing methods (Wang et al., 2018). Then consider the Morse filtration that swaps  $y$  value from  $-\infty$  to  $\infty$ . Each time the horizontal line touches the local minimum  $a < b < d < f$ , a new component that contain the local minimum is born. Each time the line touches the local maximum  $c < e$ , the two components in the sublevel set merge together. This is considered as the death of a component. Following the *Elder Rule*, when we pass a maximum and merge two components, we pair the maximum (birth) with the higher of the minimums of the two components (birth) (Edelsbrunner and Harer, 2008, 2010b; Zomorodian and Carlsson, 2005). Doing so we are pairing the birth of a component to its death. Such paired points  $(c, b)$  and  $(e, d)$  form a *persistence diagram*. In general, PD can be algebraically represented as the weighted sum of Dirac delta functions

$$I(x, y) = \sum_{i=1}^m c_i \delta(x - \xi_i, y - \tau_i) \quad (7)$$

with  $\sum_{i=1}^m c_i = 1$  such that  $I(x, y)$  becomes a 2D probability distribution satisfying

$$\int_{0 < x} \int_{y > x} I(x, y) dx dy = 1.$$

Such probabilistic representation enables us to access the vast library of distance and similarity measures between probability distributions. Given 1D functional signal  $f$  observed at

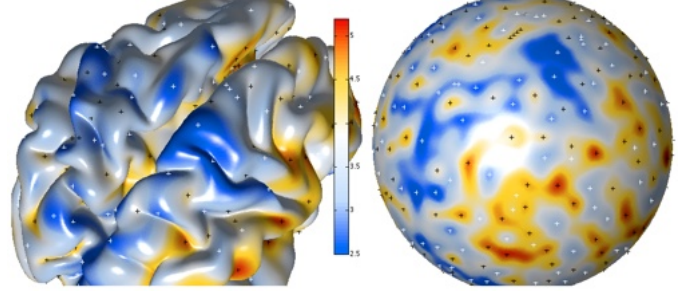


Figure 6: Heat kernel smoothing was performed on cortical thickness to obtain smooth Morse function and projected onto a sphere (Chung et al., 2009b). The critical points are identified by comparing cortical thickness around each mesh vertex. The white (black) crosses are local minimums (maximums). They will be paired following Elders rule to obtain the reduced PD.

time points  $t_1, \dots, t_n$ , the critical points can be numerically estimated obtained by checking sign changes in the finite differences  $f(t_{i+1}) - f(t_i)$ . The estimation of critical points in higher dimensional functions are done similarly by checking the signs of neighboring voxels or nodes (Figure 6) (Osher and Fedkiw, 2003; Chung et al., 2009a).

For higher dimensional Morse functions, saddle points can also create or merge sublevel sets so we also have to be concerned with them. Since there is no saddle points in 1D Morse functions, we do not need to worry about saddle points in 1D functional signals. The addition of the saddle points makes the construction of the persistence diagrams much more complex. However, the saddle points do not yield a clear statistical interpretation compared to local minimums and maximums. In fact, there is no statistical methods or applications developed for saddle points in literature. Thus, they are often removed in the *reduced Morse filtration* (Chung et al., 2009b; Pachauri et al., 2011).

The use of critical points and values within image analysis and computer vision has been relatively limited so far, and typically appear as part of simple image processing tasks such as feature extraction and edge detection (Antoine et al., 1996; Cootes et al., 1993; Sato et al., 1998). The first or second order image derivatives may be used to identify the edges of objects to serve as the contour of an anatomical shape. In this context, image derivatives are computed after image smoothing and thresholded to obtain edges and ridges of images, that are used to identify voxels likely to lie on boundaries of anatomical objects. Then, the collection of critical points are used as a geometric feature that characterize anatomical shape. Specific properties of critical values as a topic on its own, however, has received less attention so far. One reason is that it is difficult to construct a streamlined linear analysis framework using critical points or values. In brain imaging, on the other hand, the use of extreme values has been quite popular in the context of multiple comparison correction using the random field theory (Worsley et al., 1996; Taylor and Worsley, 2008; Kiebel et al., 1999). In the random field theory, the extreme value of a statistic is obtained from, and is used to compute the  $p$ -value for correcting for correlated noise across neighboring voxels.



#### 4. Why statistical analysis in TDA hard?

The persistent diagrams and equivalent barcodes are the original most often used descriptors in persistent homology. However, due to the heterogeneous nature of PD, statistical analysis of PD and barcodes have been very difficult. It is not even clear how to average PDs, the first critical step in building statistical frameworks. Even the Fréchet mean is not unique, rendering it a challenging statistical issue to perform inference on directly on PDs (Bubenik, 2015; Chung et al., 2009a; Heo et al., 2012). To remedy the problem various methods including persistent landscape (Bubenik, 2015) and persistence images are proposed.

##### 4.1. Accumulating barcodes

Even though barcodes possess similar stability properties as PD (Bauer and Lesnick, 2013), the statistical analysis of barcodes is difficult. The difficulty is due to the heterogeneous algebraic representation as an unordered multiset of intervals

$$B = \{(\xi_1, \tau_1), (\xi_2, \tau_2), \dots, (\xi_m, \tau_m)\},$$

where  $\xi_i$  and  $\tau_i$  represent the birth and death times of the  $i$ -th topological features. Barcodes may not match across different networks. Also performing averaging, and subsequently constructing test statistics is more difficult as there is no unique Fréchet mean in the space of barcodes (Kališnik, 2019; Adcock et al., 2013; Hofer et al., 2019; Turner et al., 2014).

Another important aspect is the inability of the barcode representation, and persistent homology in general, to distinguish between topological features that are intrinsic to the data or noises (Blumberg et al., 2014). Consider a point cloud data coming from some measurements (Figure 5). The four points on top are all clustered together. On the other hand, one measurement (red point) can be far from the cluster as the result of measurement error. The noisy point produces a connected component with large persistence making the barcodes ( $B_2$ ) very different from the barcodes ( $B_1$ ) from the data with no measurement error.

Due to all these limitations, barcodes are often made into summary statistics such as the minimum, maximum and average length of a collection of barcodes (Pun et al., 2018). The *accumulated persistence function* (APF) accumulates the sum of the barcodes into a scalar value:

$$APF(t) = \sum_{i: \mu_i \leq t} (\tau_i - \xi_i),$$

where  $\mu_i = (\tau_i + \xi_i)/2$  is the center of the  $i$ -th bar. Since APF is monotonically increasing over parameter  $t$ , it may be possible to construct KS-distance between APF and perform the exact topological interference (Chung et al., 2017b). APF was used in the analysis of brain artery trees (Biscio and Møller, 2019) and determining if rs-fMRI is topologically stationary over time (?).

Barcodes can also be quantified in terms of their entropy  $E(B)$ , which is another summary statistic (Atienza et al., 2018). Let  $l_i = \tau_i - \xi_i$  be the length of  $i$ -th barcode and  $L = \sum_i l_i$  be the total length of all the barcodes. Define the fraction of length

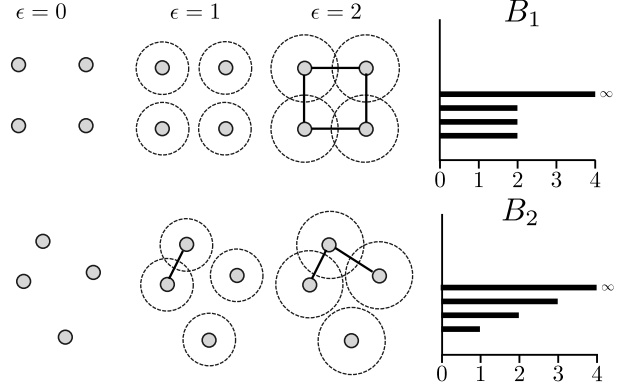


Figure 7: A comparison of the barcodes for an ordered set of points (top) and a relatively unordered set of points (bottom). In the structured system, we find that most components have equal birth ( $\xi_i$ ) and death ( $\tau_i$ ) times, which result in equal lifetimes for each topological feature ( $\tau_i - \xi_i$ ), resulting in a high persistence entropy  $E(B_1)$ . For unordered system  $B_2$ , the components have different death points. Thus unordered system will have a lower persistence entropy, i.e.,  $E(B_1) > E(B_2)$ .

of the  $i$ -th barcode  $p_i = l_i/L$  over the total length. Then the entropy  $E(B)$  is defined as

$$E(B) = - \sum_i p_i \log p_i \quad (8)$$

This provides a single scalar measurement of the topological disorder of a given system (Atienza et al., 2018). The barcode entropy has also been shown to outperform graph theoretic entropies, such as connectivity entropy and Von Neumann entropy when characterizing complex networks and has been used in identifying pre-ictal and ictal states in epilepsy patients (Rucco et al., 2016; Merelli et al., 2015, 2016).

Figure 7 illustrates how entropy differs for two systems. For ordered system  $B_1$ , ignoring the component that dies at  $\infty$ , we have  $L = 6$ ,  $p_1 = p_2 = p_3 = 2/6$  and  $E(B_1) = 1.1$ . For unordered system  $B_2$ , we have  $L = 6$ ,  $p_1 = 1/6$ ,  $p_2 = 2/6$ ,  $p_3 = 3/6$  and  $E(B_2) = 1.0$ . Thus, ordered system has higher persistent entropy  $E(B_1) > E(B_2)$ . It would be interesting to investigate if brain networks will exhibit lower persistence entropy.

##### 4.2. Persistent landscape

The PD and equivalent barcodes are the original most often used descriptors in persistent homology. Even though, the PD possess desirable bounding properties such as Lipschitz stability with respect to the bottleneck distance, statistical analysis on persistent PD is difficult since it is not a clear cut to define the average PD. Further the Fréchet mean is not unique in PD, rendering it a challenging statistical issue to perform inference directly on PD (Chung et al., 2009a; Heo et al., 2012). To remedy the problem, persistent landscape was first proposed in Bubenik (2015), which maps PD to a Hilbert space.

Given a barcode  $(\xi_i, \tau_i)$ , we can define the piecewise linear bump function  $h_i$  as

$$h_i(\epsilon) = \max(\min(\epsilon - \xi_i, \tau_i - \epsilon), 0). \quad (9)$$

The geometric representation of the bump function (9) is a right-angled isosceles triangle with height equal to half of the base of

the corresponding interval in the barcode (Wang et al., 2019; Bubenik, 2020). Given barcodes  $(\xi_1, \tau_1), \dots, (\xi_m, \tau_m)$ , the persistent landscape  $\lambda_k(\epsilon)$  is defined as

$$\lambda_k(\epsilon) = k\text{-th largest value of } h_i(\epsilon).$$

With this new representation, it is possible to have a unique mean landscape  $\bar{\lambda}(\epsilon)$  over subjects, which is simply the average of persistent landscape at each filtration value  $\epsilon$ . Such operation enables to construct the usual test statistic. The persistent landscapes have been applied to epilepsy EEG (Wang et al., 2018, 2019), functional brain networks in fMRI (Stolz et al., 2017, 2018) and neuroanatomical shape modeling in MRI (Garg et al., 2017). However, since scatter points in PD are converted to piecewise linear functions, statistical analysis on persistent landscape is not necessarily any easier than before. Thus, additional transformations on persistent landscapes is needed for statistical analysis. Since  $\lambda_1(\epsilon) > \lambda_2(\epsilon) > \lambda_3(\epsilon) > \dots$  at each fixed  $\epsilon$ , it is possible to build a monotone sequence of scalar values by computing area under function  $\lambda_k(\epsilon)$  and perform the exact topological inference (Wang et al., 2019).

#### 4.3. Smoothing PD

Due to the heterogenous nature of persistent diagrams (PD), it is difficult to even average PDs and perform statistical analysis. Starting with Chung et al. (2009a), various smoothing methods were applied to PD such that averaging and statistical analysis can be directly performed on the smoothed PD. Chung et al. (2009a) discretized PD using the uniform square grid. A concentration map is then obtained by counting the number of points in each pixel, which is equivalent to smoothing PD with a uniform kernel. Notice that this approach is somewhat similar to the voxel-based morphometry (Ashburner and Friston, 2000), where brain tissue density maps are used as a shapeless metric for characterizing concentration of the amount of tissue. In Reininghaus et al. (2015), persistence scale-space kernel approach was proposed, where points in PD is treated as heat sources modeled as Dirac-delta and diffusion is run on PD. Ignoring symmetry across  $y = x$ , diffusion of heat sources is given by the convolution of Gaussian kernel

$$K_\sigma(x, y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/(2\sigma^2)}$$

on Dirac-delta functions as

$$f(x, y) = \sum_{i=1}^m K_\sigma * \delta(\xi_i, \tau_i) = \sum_{i=1}^m \frac{1}{2\pi\sigma^2} e^{-[(x-\xi_i)^2 + (y-\tau_i)^2]/(2\sigma^2)}.$$

This representation is known as the *persistence surface* (Adams et al., 2017). Heat diffusion  $f(x, y)$  is then discretized by integrating the function within each square grid of given resolution. An example of the discretization is given in Figure 8-c. A *persistence image* (PI) is simply the vector of square grid values. There exists a stability result bounding the  $L_2$ -distance between PI by the bottleneck distance  $D_B$  or Wasserstein distance  $D_W$  on PD (Adams et al., 2017):

$$\|PI - PI'\|_2 \leq L \cdot D(PD, PD') \quad (10)$$

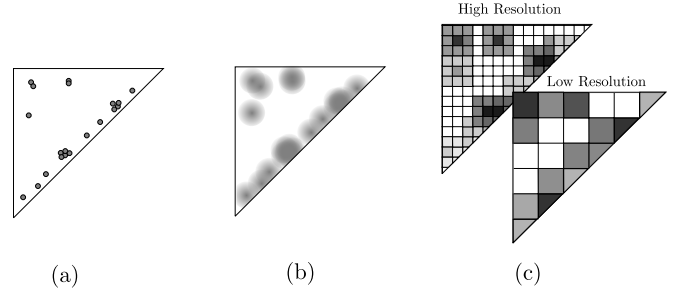


Figure 8: (a) Persistence Diagram (PD) scatter plot. (b) Persistence Surface. (c) Persistence Image (PI) at multiple grid resolutions. An illustration of the process for converting a PD into a PI representation. The scatter plot presented in (a) is fit with multiple 2-dimensional Gaussians centered at each point in the PD, known as the persistence surface (b). The persistence surface is then discretized into a persistence image (c) by overlaying a grid, and assigning each grid square a scalar value equal to the integral of the sum of the Gaussians within the bounds of the grid square.

for some constant  $L$ . This is consequence of kernel smoothing, which reduces the spatial variability. The vectorized representation enables the PI to be used in various statistical and machine learning applications (Chazal and Michel, 2017). PI was used in the functional brain network study on Schizophrenia (Stolz et al., 2018) and in conjunction with deep learning for autism classification (Rathore et al., 2019). The major theme across these applications is that the PI presents a simple, vectorized representation of the persistence diagram that can be readily used in many statistical and machine learning algorithms (Pun et al., 2018). The PI can be easily manipulated by adjusting the size of grid as well as the bandwidth  $\sigma$  of kernel. This allows users to adjust the amount of information needed to represent PD in multiple scales for application needs.

One major drawback of the method is the inherent loss of information from both the grid discretization and the smoothing of PD. This loss of information can be detrimental to the analysis of the given dataset as important features may be over smoothed or unintentionally grouped together. One method of combatting this potential loss of information is to utilize weighting function  $w(\xi, \tau)$  on the PI representation to emphasize certain features (Divol and Polonik, 2019; Adams et al., 2017). A popular method used is to assign linear or exponential weight such as  $w(\xi, \tau) = (\tau - \xi)$  for each point in the PD that increases in correlation with the distance from the diagonal of the persistence diagram before diffusion via the Gaussian kernel. This weights more toward the features of higher persistence making them robust to the subsequent analysis.

## 5. Big data, scalability and approximation

### 5.1. Computational complexity

The huge volumes of data in various imaging fields including neuroimaging and cosmology necessitate efficient algorithms for performing scalable TDA computation. The computing needs in these diverse fields share a lot of similarities and we can adapt many scalable methods developed in other areas to brain

imaging data. In this section, we review some of the scalable schemes for simplifying big data computations. There are several ways in which TDA can help with big data challenges. One of which is to dimensionality reduction.

Given  $n$  scatter points data in a  $k$  dimensional space, the amount of information grows exponentially as  $k^n$ . In contrast, persistent homology compresses the data to  $k$  functions  $f_0(\epsilon)$ ,  $f_1(\epsilon)$ ,  $\dots$ ,  $f_{k-1}(\epsilon)$  of filtration value  $\epsilon$ . Topological features such as the duration of birth to death of  $k$ -cycles, Betti numbers and the number of such  $k$ -cycles are used for such functions. If we further bin the filtration parameter into  $N_{bin}$  intervals, the amount of information characterized by persistent homology grows only linearly with  $k$  as  $kN_{bin}$ . This huge compression of data can facilitate the search for its underlying patterns in big data. Even so, performing TDA on large dataset is daunting. Persistent homology does not scale well with increased network size (Figure 9). The computational complexity of persistent homology grows rapidly with the number of simplices (Topaz et al., 2015). With  $n$  nodes, the size of the  $k$ -skeleton grows as  $n^{k+1}$ . Homology calculations are often done by Gaussian elimination, and if there are  $N_{simplices}$  simplices, it takes  $O(N_{simplices}^3)$  time to perform. In a  $d$  dimensional embedding space, this computational time can be estimated as  $O(n^{3k+3})$  (Solo et al., 2018). Thus, a full TDA computation of Rips complex is exponentially costly; it becomes infeasible when the number of nodes or the dimension of the embedding space is large. This can easily happen when one tries to use brain networks at the voxel level resolution. Thus, there have been many algorithm development in computing Rips complex approximately but fast for large-scale data. There are two broad approaches to make the TDA computation scalable: 1) by sampling a subset of the nodes and 2) by constructing alternate filtrations consisting of significantly smaller number of simplices while keeping the set of nodes unchanged that give the same or approximate persistent homologies of the filtration. This section is focused on the detailed review of the first approach. The construction of alternate filtrations such as graph filtrations are reviewed in the next section.

The second approach speeds up the computation by reducing the number of simplices. The alpha filtration based on Delaunay triangulation is an example of an alternate filtration with a provable guarantee that it consists of smaller number of simplices than the Rips filtration (Otter et al., 2017). The worst-case number of simplices in the alpha filtration in dimension  $k = 3$  scales as  $O(n^2)$ . The benefits of a sparser representation than Rips filtration cannot be overstated, but the challenge is to construct a simplified filtration without spoiling the stability results of persistent homology. By perturbing the metric of the data space, one can construct a sparsed Rips filtration which has  $O(n)$  simplices and can be computed in  $O(n \log n)$  time (Sheehy, 2013).

## 5.2. Witness complex

One crucial difference between geometry and topology is that topology can be accurately estimated from a small sample of the full data, while geometry requires more fine-grained information. This reflects the fact that topology is a more basic

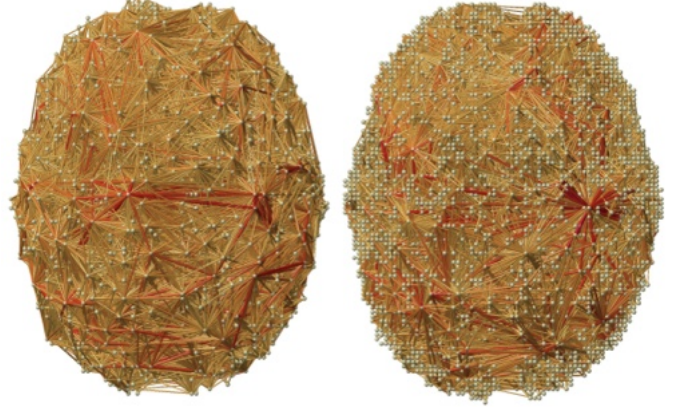


Figure 9: rs-fMRI correlation network of two subjects from HCP with more than 25000 nodes. Identifying cycles and computing the number of cycles can be computationally demanding in this type of dense correlation network since persistent homology computations are not very scalable.

notion of shape than geometry. This suggests a sampling procedure can significantly reduce the size of a simplicial representation of data while preserving topological information. Since the number of simplices in the Rips complex grows with respect to the number of vertices  $N_{vertices}$  as  $2^{O(N_{vertices})}$ , being able to evaluate topology on a small sample of the vertices can greatly speed up the computation. A principled procedure for sampling data points to identify the data's topology is given by *witness complexes* (de Silva and Carlsson, 2004). The idea is to use a small subset of the given point cloud as landmarks, which form the vertices of the complexes. The remaining non-landmark points are used as witnesses to the existence of simplices spanned by combinations of landmark points.

Let  $Z$  be the full point cloud,  $L \subset Z$  denote the set of landmark points, and  $W \subset Z$  denotes the set of witnesses.  $W$  is a dense sample of  $Z$ . Then  $w \in W$  is a *weak witness* of a simplex  $\sigma = [v_0, v_1, \dots, v_k] \subset L$  if for all  $v \in \sigma$  and for all  $v' \in L \setminus \sigma$ ,  $d(w, v) \leq d(w, v')$  (de Silva and Carlsson, 2004). The largest distance from a witness to the simplex should be smaller than the distance to all other points excluding the simplex. A  $k$ -simplex is witnessed by  $w$  if it consists of  $w$ 's  $(k+1)$ -th nearest neighbors in  $Z$ . Let  $d_i(w, \sigma)$  is the  $(i+1)$ -th closest distance from  $w$  to one of vertices in  $\sigma$ . Note  $d_k(w, \sigma) = \max_k d(w, v_k)$ . Similarly define  $d_k(w, L)$  to be the distance from  $w$  to its  $(k+1)$ -th closest landmark point in  $L$ . Starting with initial random point  $v_0$ , the witness complex can be defined by induction. For vertices  $v_i \in L$ , we include the  $k$ -simplex  $\sigma$  in the witness complex  $\text{Wit}(L, W, \epsilon)$  at scale  $\epsilon$  if all of its faces belong to  $\text{Wit}(L, W, \epsilon)$  and there exists a witness point  $w \in W$  such that

$$d_k(w, \sigma) \leq \epsilon + d_k(w, L) \quad (11)$$

for filtration value  $\epsilon$  which is analogous to the filtration values in the Rips filtration. The witness complex is the largest simplicial complex with vertices in  $L$  whose faces are witnessed by points in  $W$  (Guibas and Oudot, 2008).

The witness complex depends on the initial choice of landmarks. We want to minimize the number of landmark points



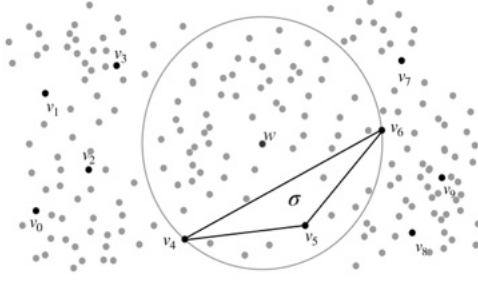


Figure 10: In the witness complex, edges and higher-dimensional simplices are built out of vertices in the landmark set  $L = \{v_0, v_1, \dots, v_9\}$  and witnessed by other points according to inequality (11). Point  $w$  is a weak witness of the 2-simplex  $[v_4, v_5, v_6]$  at filtration scale  $\epsilon = 0$ . This 2-simplex belongs to the witness complex  $\text{Wit}(L, W, 0)$  since  $\sigma = [v_4, v_5, v_6]$  satisfies (11) for  $k = 2$  and all its faces (edges)  $[v_4, v_5]$ ,  $[v_5, v_6]$ ,  $[v_6, v_4]$  satisfy (11) for  $k = 1$ .

to reduce computational time, but yet this landmark set should be representative of the data, as the four points in representing the circle in Figure 10. The landmarks are chosen either randomly or through the *maxmin* algorithm, which tend to select evenly spaced landmarks (de Silva and Carlsson, 2004; Cole and Shiu, 2019). In brain imaging applications, the landmark can be chosen biologically. In building parcellation-based networks, center of individual parcellation can be chosen as landmarks and approximate the overall topology of large-scale brain networks constructed at the voxel-level. The size of a witness complex is determined by the size of the landmark set  $L$ . Since  $L$  represents only a fraction of the points in given point cloud  $Z$ , witness complexes are much smaller than the Rips complexes corresponding to  $Z$ . The upper bound  $|Z|/|L| \leq 20$  was suggested for 2D surface data (de Silva and Carlsson, 2004). Taking  $|Z|/|L| = 20$  and denoting by  $N_W, N_R$  the sizes of the corresponding witness complex and Rips complex respectively, the worst-case scaling of the number of simplices in the witness complex is (Otter et al., 2017; Arafat et al., 2019)

$$N_W \sim 2^{|L|} \sim 2^{|Z|/20} \sim N_R^{1/20}.$$

Therefore the witness complex presents a huge computational advantage for large data sets.

Witness complexes can be thought of as approximations of the Delaunay triangulation in 2D (de Silva and Carlsson, 2004; Attali et al., 2007; Guibas and Oudot, 2008; Boissonnat et al., 2009b). The advantage of the witness complex is that the ambient dimension does not affect the complexity of the algorithm, while the Delaunay triangulation suffers from the curse of dimensionality. An algorithm was proposed in Boissonnat et al. (2009b) to enrich the set of witnesses and landmarks to preserve the relation between the Delaunay triangulation and the witness complex in  $k > 2$  dimensions so as to ensure that the witness complex is a good approximation of the underlying topology of data.

Witness complexes were applied in the study of natural images and identifying them with the topology of the Klein bottle (Carlsson et al., 2008) and the study the topology of activity patterns in the primary visual cortex (Singh et al., 2008). Interestingly, spontaneous activation patterns and activity stimu-

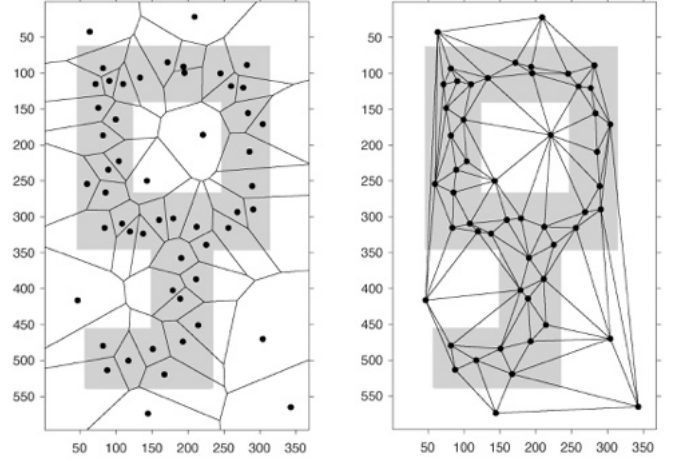


Figure 11: Left: The Voronoi diagram obtained from the point cloud that samples the key shaped object. Right: The Delaunay complex, which is the dual graph of the Voronoi diagram.

lated by natural images were found to have compatible topological structure, that of a two-sphere. (Dabaghian et al., 2012) used witness complexes to model activity in the hippocampus as storing topological information about the local environment.

### 5.3. $\alpha$ -filtrations

The  $\alpha$ -filtration (Edelsbrunner and Mücke, 1994) takes its inspiration from often used Delaunay triangulations, which is a simplicial complex. The Delaunay triangulation or tetrahedralization has been widely used in brain imaging field for building surface or volumetric meshes from MRI (Si, 2015; Wang et al., 2017a).

Given a set of points  $X = \{x_1, \dots, x_p\} \subset \mathbb{R}^D$ , the *Voronoi cell* around  $x_k$  is given by

$$V_k = \{y \in \mathbb{R}^D \mid d(y, x_k) \leq d(y, x_j) \text{ for all } j \neq k\}.$$

The Voronoi cell of point  $x_k$  includes all points for which  $x_k$  is the nearest point among all points in  $X$ . The Voronoi cell is a convex set. The collection of Voronoi cells triangulates the domain  $\mathbb{R}^D$  and constitutes the *Voronoi diagram* of  $X$ . The *Delaunay complex* is defined via the Voronoi diagram as  $\cup_k V_k$ . An example of this geometric realization is shown in Figure 11. The Delaunay triangulation is then defined as the dual graph of the Voronoi diagram. The minimum spanning tree of the complete graph consisting of node set  $X$  and edge weights  $d(x_k, x_j)$  is a subset of the Delaunay triangulation.

The size of the Delaunay complex grows with the number of vertices  $N_{\text{vertices}}$  as  $N_{\text{vertices}} \log N_{\text{vertices}}$  for  $D \leq 2$  and  $N_{\text{vertices}}^{D/2}$  for  $D \geq 3$  (Otter et al., 2017). This is a clear advantage of the Delaunay complex over the Rips and Čech complexes whose size grow exponentially with the number of vertices, though the Delaunay complex still suffers from the curse of dimensionality. Developing efficient algorithms for handling higher dimensional Delaunay complex is a subject of ongoing research (Boissonnat et al., 2009a). Wang et al. (2017b) used the Delaunay triangulation in building a complex from EEG channel locations.



To further speed up the computation,  $\alpha$ -complex is often used over Rips complex. The  $\alpha$ -complex, which is a subcomplex of the Delaunay complex, is defined as follows. For each point  $x \in X$  and  $\epsilon > 0$ , let  $B_x(\epsilon)$  be the closed ball with center  $x$  and radius  $\epsilon$ . The union of these balls  $\cup_{x \in X} B_x(\epsilon)$  is the set of points in  $\mathbb{R}^D$  at distance at most  $\epsilon$  from at least one of the points of  $X$ . We then intersect each ball with the corresponding Voronoi cell,  $R_x(\epsilon) = B_x(\epsilon) \cap V_x$ . The collection of these sets  $\{R_x(\epsilon) \mid x \in X\}$  forms a cover of  $\cup_{x \in X} B_x(\epsilon)$ , and the  $\alpha$ -complex of  $X$  at scale  $\epsilon$  is defined as

$$\text{Alpha}(X, \epsilon) = \{\sigma \subset X \mid \cap_{x \in \sigma} R_x(\epsilon) \neq \emptyset\}.$$

Since balls and Voronoi cells are convex, the  $R_x(\epsilon)$  are also convex.  $\text{Alpha}(X, \epsilon)$  has the same homology as  $\cup_{x \in X} B_x(\epsilon)$ . Not only is  $\text{Alpha}(X, \epsilon)$  a subcomplex of the Delaunay complex, it is also a subcomplex of the Čech complex.

One potentially undesirable aspect of the  $\alpha$ -filtration is that the topological objects it identifies die at smaller scales than one would expect when there are outliers. Thus, the  $\alpha$ -filtration is sensitive to outliers. We can try to remedy this situation by modifying the  $\alpha$ -filtration to a weighted  $\alpha$ -filtration that allows balls of different sizes, and building the filtration in a spatially adaptive fashion. In applications, this weighted  $\alpha$ -filtration is often motivated by the underlying physical and biological models. For example, in studying the persistent homology of large-scale structure data in cosmology (Biagetti et al., 2020), the gravitational potential depends on the mass density of clusters (0-simplices) suggesting a varying ball sizes. In modeling of biomolecules, such as proteins, RNA, and DNA, each atom is represented by a ball whose radius reflects the range of its van der Waals interactions and thus depends on the atom type. For brain network applications, we can adjust the size of balls to follow the contour of either gray matter or white matter of the brain and adaptively build the filtration.

A different modification to the  $\alpha$ -filtration can be done that reduces the effect of outliers (Biagetti et al., 2020). In  $\alpha$ DTM-filtration, we assign the simplices a *filtration time* based on the distance-to-measure (DTM) function. Given a set of point  $X$ , the empirical DTM function is given by

$$\text{DTM}(x) = \frac{1}{k} \left( \sum_{x_i \in N_k(x)} \|x - x_i\|^p \right)^{1/p} \quad (12)$$

where  $N_k(x)$  is the list of the  $k$  nearest neighbors in  $X$  to  $x$  (Chazal et al., 2011). Often  $p = 2$  is used. One can build intuition by considering the case  $k = 1$ , in which the DTM function gives the distance to the nearest point in  $X$ . Increasing  $k$  “smooths” this distance function so that it takes small values where it is near many points and large values near outliers. Then performing a sublevel filtration using the DTM function, outliers will be added relatively late in the filtration, leading to a smaller effect on the persistent homology.

For large-scale images, DTM function can be evaluated on an image grid (Xu et al., 2019). Evaluating on a grid with high resolution and sufficiently large  $k$  involves prohibitive computational expense. One way to get around this is to evaluate

the DTM function at certain relevant points of the space. This amounts to choosing an efficient triangulation of the ambient space. In Biagetti et al. (2020), the DTM function was evaluated on the Delaunay complex. This was shown to keep the computational cost low while maintaining the desirable feature of  $\alpha$ DTM-filtration in reducing the adverse effects of outliers.

#### 5.4. Combinatorial Morse theory

Another approach to significantly reduce the number of simplices is Morse filtration. Using the combinatorial Morse theory (Forman, 1998), one can reduce the initial complex using geometric and combinatorial methods before performing homology calculations. Combinatorial Morse theory extends the notion of Morse homology for smooth manifolds to discrete datasets. The gradient flow for a smooth manifold is replaced by a partial pairing of cells in a complex. As a result, the Morse complex has thus a significantly smaller size than the original complex, while preserving all homological information. As demonstrated in Harker et al. (2010), for many complexes the resulting Morse complex is many orders of magnitude smaller than the original. An efficient preprocessing algorithm was developed in Mischaikow and Nanda (2013) that extends combinatorial Morse theory from complexes to filtrations. This preprocessing algorithm maps any filtration into a Morse filtration with the same persistent homologies, thus significantly reducing both the computational time and the required memory for running the persistence algorithm.

#### 5.5. Hypergraphs

The computational overhead in TDA lies in the homology calculations of higher-dimensional simplices. The primary algorithms in persistent homology requires Gaussian elimination with cubic runtime in the number of simplices (Edelsbrunner and Harer, 2010b). This can cause an additional computational bottleneck when the number of simplices up to dimension  $k$  in the Rips filtration is the order of  $O(n^{k+1})$  for  $n$  data points (Sheehy, 2013). Thus, it is worthwhile to explore if the run time for analyzing data structure is significantly reduced if we represent the nodes and their relations in terms of *hypergraphs* (Torres et al., 2020). A hypergraph is a generalization of a graph which contains vertex set  $V$  and hyperedge set  $E$  consisting of vertices allowing polyadic relations between an arbitrary number of vertices.

This is in contrast to the normally defined graph which can only form connections between two vertices. Unlike an edge in a graph or a simplicial complex, a hyperedge can connect any arbitrary number of vertices. Thus, a hypergraph representation of data consists of only 0D and 1D objects. Figure 12 displays a hypergraph with three nodes  $v_1, v_2, v_3$  representing interactions between all three nodes with hyperedge  $e_1$ .

We can represent the hypergraph using the incidence matrices or the corresponding adjacency matrices. The incidence matrix  $H = (h_{ij})$  is a matrix with rows representing vertices  $v_i$  and columns representing edges  $e_j$  (Estrada and Rodriguez-

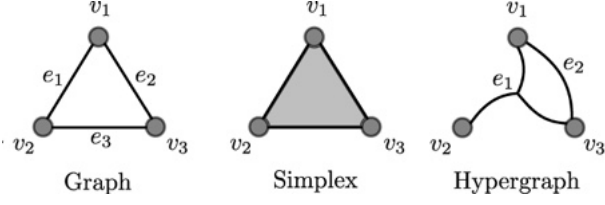


Figure 12: Schematic of graph and hypergraph with three nodes  $v_1, v_2, v_3$ . The edge set for the graph is  $e_1 = [v_1, v_2]$ ,  $e_2 = [v_1, v_3]$ ,  $e_3 = [v_2, v_3]$ . The edge set for the hypergraph is given by  $e_1 = [v_1, v_2, v_3]$ ,  $e_2 = [v_1, v_3]$ . A hypergraph allows edges that contain an arbitrary number of vertices, while edges in a graph is limited to only two vertices, preventing it from capturing higher level interactions between vertices. The connection  $e_1$  in the hypergraph captures the high level interaction between the three vertices. A graphs on the other hand can only approximate the higher order interaction with a series of pairwise interactions. The simplex captures the high level interaction but also assumes pairwise interactions between all vertices. The hypergraph captures the high level interaction directly.

(Velazquez, 2005) defined as

$$h_{ij} = \begin{cases} 1 & v_i \in e_j \\ 0 & v_i \notin e_j \end{cases}.$$

The adjacency matrix  $A$  for a given graph or hypergraph is derived from the incidence matrix

$$A = HH^T - D,$$

where  $D$  is the diagonal matrix whose diagonal entries are the degrees of each node (Estrada and Rodriguez-Velazquez, 2005). The adjacency matrix of a hypergraph can be used to compute certain properties of hypergraphs such as centrality and clustering coefficients (Estrada and Rodriguez-Velazquez, 2005). The incidence matrix  $H$  and adjacency matrix  $A$  of the graph in Figure 12 is given by

$$H = \begin{matrix} & e_1 & e_2 & e_3 \\ \begin{matrix} v_1 \\ v_2 \\ v_3 \end{matrix} & \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} \end{matrix}, \quad A = \begin{matrix} & v_1 & v_2 & v_3 \\ \begin{matrix} v_1 \\ v_2 \\ v_3 \end{matrix} & \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} \end{matrix}.$$

Similarly, the incidence and adjacency matrix of the hypergraph in Figure 12 is given by

$$H = \begin{matrix} & e_1 & e_2 \\ \begin{matrix} v_1 \\ v_2 \\ v_3 \end{matrix} & \begin{pmatrix} 1 & 1 \\ 1 & 0 \\ 1 & 1 \end{pmatrix} \end{matrix}, \quad A = \begin{matrix} & v_1 & v_2 & v_3 \\ \begin{matrix} v_1 \\ v_2 \\ v_3 \end{matrix} & \begin{pmatrix} 0 & 1 & 2 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{pmatrix} \end{matrix}.$$

Simplicial complexes are also capable of capturing higher level interactions, but require representation of higher order relationships using high-dimensional simplices and have the restriction that these relationships have a property known as the *downward inclusion* (Torres et al., 2020). The downward inclusion requires that any subset of  $(k + 1)$ -vertices that form a  $k$ -simplex must also form a simplex. This reduces the ability of the simplicial complex to represent abstract relationships, such as those used to understand the cross link connectivity of functional brain networks over time (Bassett et al., 2014), as

it assumes the presence of lower level connections via downward inclusion. It is possible to have a high order interaction between three regions of the brain without significant pairwise interaction between any two regions. Thus, the simplicial complex based brain network analysis may over enforce additional low hierarchical dependency than needed.

In Figure 12 we illustrate the difference between a graph, a simplex, and a hypergraph in capturing a high level relationship between three vertices. The graph can only capture dyadic relationship between two nodes. For higher order interaction between all three nodes  $v_1, v_2, v_3$ , additional model is needed. The simplex represents the higher level relationships between all three vertices by the filled-in triangle but requires that all dyadic connections between all node pairs exist, which may not necessarily be true in brain networks. However, the hypergraph captures only the triadic relationship between the vertices without lower level dyadic relationship.

A hypergraph representation of data consists of only 0D (vertices) and 1D (hyperedges) objects making it a low dimensional representation of higher order interactions within a dataset. Hypergraphs provide improved flexibility over simplicial representations used in TDA and accurately capture higher level relationships in complex networks, such as those used to understand covariant structures in neurodevelopment (Torres et al., 2020; Gu et al., 2017). The hypergraph has found many applications in functional brain imaging studies. Hypergraph topology has been used to differentiate brain patterns during attention-demanding tasks, memory-demanding tasks, and resting states (Davison et al., 2015). Hypergraph topology has also been used in understanding differences in functional connectivity in brains over the human lifespan (Davison et al., 2016). It has also been used in the diagnosis of brain disease and in the identification of connectome biomarkers (Jie et al., 2016; Zu et al., 2016). The use of hypergraphs extends beyond functional networks where they have been used in multi-atlas and multi-modal hippocampus segmentation by developing hypergraphs that represent topological connections across multiple images over different imaging modalities (Dong et al., 2015).

Many of the reported methods in literature use simple topological descriptors of hypergraphs, such as number of cross links, hyperedge sizes or clustering coefficients. There has been little application of the concepts of TDA to hypergraphs directly. However, there has been work some preliminary works connecting TDA to hypergraphs including the the analysis of Betti numbers of hypergraphs, the cohomology of hypergraphs, and the embedded homology of hypergraphs (Chung and Graham, 1992; Emtander, 2009; Bressan et al., 2016).

The application of TDA methods to hypergraphs may yield faster running times as the homology of high dimensional simplices are no longer needed to represent high order interactions, and could result in more meaningful analysis as constraints such as downward inclusion would no longer be required (Lloyd et al., 2016).

## 6. Persistent homology in brain networks

For adapting persistent homology to brain network data, *graph filtration* was introduced (Lee et al., 2011a, 2012). Instead of analyzing networks at one fixed threshold, we build the collection of nested networks over every possible threshold. The graph filtration is a threshold-free framework for analyzing a family of graphs but requires hierarchically building nested subgraph structures. The graph filtration framework shares similarities to existing multi-thresholding or multi-resolution network models that use somewhat arbitrary multiple thresholds or scales (Achard et al., 2006; He et al., 2008; Kim et al., 2015; Lee et al., 2012). However such approaches are mainly exploratory and mainly used to visualize graph feature changes over different thresholds without quantification. Persistent homology, on the other hand, quantifies such feature changes in a coherent way.

### 6.1. Graph filtration

The graph filtration has been the first type of filtrations applied in brain networks and now considered as the de fact baseline filtrations in the field (Lee et al., 2011a, 2012). Euclidean distance is often used metric in building filtrations in persistent homology (Edelsbrunner and Harer, 2010b). Most brain network studies also use the Euclidean distances for building graph filtrations (Lee et al., 2011b, 2012; Chung et al., 2015a, 2017b; Petri et al., 2014; Khalid et al., 2014; Cassidy et al., 2015; Wong et al., 2016; Anirudh et al., 2016; Palande et al., 2017). Given weighted network  $X = (V, w)$  with edge weight  $w = (w_{ij})$ , the binary network  $X_\epsilon = (V, w_\epsilon)$  is a graph consisting of the node set  $V$  and the binary edge weights  $w_\epsilon = (w_{\epsilon,ij})$  given by

$$w_{\epsilon,ij} = \begin{cases} 1 & \text{if } w_{ij} > \epsilon; \\ 0 & \text{otherwise.} \end{cases} \quad (13)$$

Note Lee et al. (2011a, 2012) defines the binary graphs by thresholding above such that  $w_{\epsilon,ij} = 1$  if  $w_{ij} < \epsilon$  which is consistent with the definition of the Rips filtration. However, in brain connectivity, higher value  $w_{ij}$  indicates stronger connectivity so we usually thresholds below (Chung et al., 2015a).

Note  $w_\epsilon$  is the adjacency matrix of  $X_\epsilon$ , which is a simplicial complex consisting of 0-simplices (nodes) and 1-simplices (edges) (Ghrist, 2008). In the metric space  $X = (V, w)$ , the Rips complex  $\mathcal{R}_\epsilon(X)$  is a simplicial complex whose  $(p-1)$ -simplices correspond to unordered  $p$ -tuples of points that satisfy  $w_{ij} \leq \epsilon$  in a pairwise fashion (Ghrist, 2008). While the binary network  $X_\epsilon$  has at most 1-simplices, the Rips complex can have at most  $(p-1)$ -simplices. Thus,  $X_\epsilon \subset \mathcal{R}_\epsilon(X)$  and its complement  $X_\epsilon^c \subset \mathcal{R}_\epsilon(X)$ . Since a binary network is a special case of the Rips complex, we also have

$$X_{\epsilon_0} \supset X_{\epsilon_1} \supset X_{\epsilon_2} \supset \dots$$

and equivalently

$$X_{\epsilon_0}^c \subset X_{\epsilon_1}^c \subset X_{\epsilon_2}^c \subset \dots$$

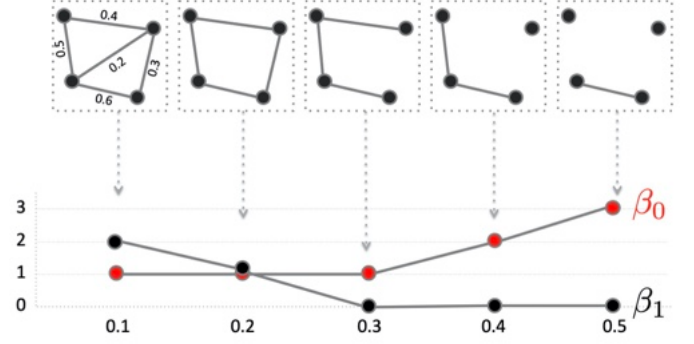


Figure 13: Schematic of graph filtration and Betti curves. We sort the edge weights in an increasing order. We threshold the graph at filtration values and obtain binary graphs. The thresholding is performed sequentially by increasing the filtration values. The 0-th Betti number  $\beta_0$ , which counts the number of connected components, and the first Betti number  $\beta_1$ , which counts the number of cycles, is then plotted over the filtration. The Betti plots curves monotone in graph filtrations.

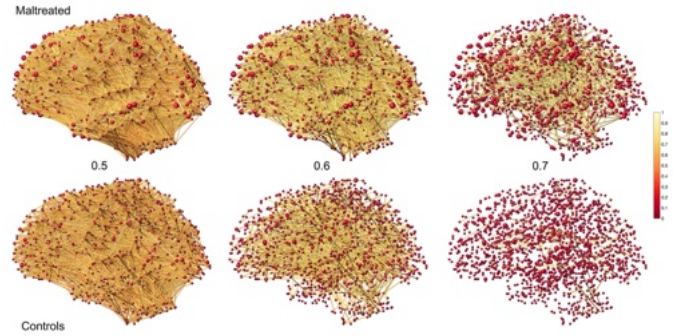


Figure 14: Graph filtrations of maltreated children vs. normal control subjects on FA-values (Chung et al., 2015a). The Pearson correlation is used as filtration values at 0.5, 0.6 and 0.7. maltreated subjects show much higher correlation of FA-values indicating more homogeneous and less varied structural covariate relationship.

for  $0 = \epsilon_0 \leq \epsilon_1 \leq \epsilon_2 \dots$ . The sequence of such nested multi-scale graphs is defined as the *graph filtration* (Lee et al., 2011a, 2012). Figure 13 illustrates a graph filtration in a 4-nodes example while Figure 14 shows the graph filtration on structural covariates on maltreated children on 116 parcellated brain regions. fv

Note that  $X_0$  is the complete weighted graph while  $X_\infty$  is the node set  $V$ . By increasing the threshold value, we are thresholding at higher connectivity so more edges are removed. Given a weighted graph, there are infinitely many different filtrations. This makes the comparisons between two different graph filtrations difficult. For network  $\mathcal{Y} = (V, z)$  with the same node set but with different edge weight  $z$ , with different filtration values,  $\lambda_0 \leq \lambda_1 \leq \lambda_2 \dots$ , we have

$$\mathcal{Y}_{\lambda_0} \supset \mathcal{Y}_{\lambda_1} \supset \mathcal{Y}_{\lambda_2} \supset \dots$$

Then how we compare two different graph filtrations  $\{X_{\epsilon_i}\}$  and  $\{\mathcal{Y}_{\lambda_i}\}$ ? For different  $\epsilon_j$  and  $\epsilon_{j+1}$ , we can have identical binary graph, i.e.,  $X_{\epsilon_j} = X_{\epsilon_{j+1}}$ . For graph  $X = (V, w)$  with  $q$  unique positive edge weights, the maximum number of unique filtrations is  $q + 1$  (Chung et al., 2015a).



For a graph with  $p$  nodes, the maximum number of edges is  $(p^2 - p)/2$ , which is obtained in a complete graph. If we order the edge weights in the increasing order, we have the sorted edge weights:

$$0 = w_{(0)} < \min_{j,k} w_{jk} = w_{(1)} < w_{(2)} < \dots < w_{(q)} = \max_{j,k} w_{jk},$$

where  $q \leq (p^2 - p)/2$ . The subscript  $(\cdot)$  denotes the order statistic. For all  $\lambda < w_{(1)}$ ,  $\mathcal{X}_\lambda = \mathcal{X}_0$  is the complete graph of  $V$ . For all  $w_{(r)} \leq \lambda < w_{(r+1)}$  ( $r = 1, \dots, q-1$ ),  $\mathcal{X}_\lambda = \mathcal{X}_{w_{(r)}}$ . For all  $w_{(q)} \leq \lambda$ ,  $\mathcal{X}_\lambda = \mathcal{X}_{p(q)} = V$ , the vertex set. Hence, the filtration given by

$$\mathcal{X}_0 \supset \mathcal{X}_{w_{(1)}} \supset \mathcal{X}_{w_{(2)}} \supset \dots \supset \mathcal{X}_{w_{(q)}}$$

is *maximal* in a sense that we cannot have any additional filtration  $\mathcal{X}_\epsilon$  that is not one of the above filtrations. Thus, graph filtrations are usually given at edge weights.

The condition of having unique edge weights is not restrictive in practice. Assuming edge weights to follow some continuous distribution, the probability of any two edges being equal is zero. For discrete distribution, it may be possible to have identical edge weights. Then simply add Gaussian noise or add extremely small increasing sequence of numbers to  $q$  number of edges.

## 6.2. Monotone Betti curves

The graph filtration can be quantified using monotonic function  $f$  satisfying

$$f(\mathcal{X}_{\epsilon_0}) \geq f(\mathcal{X}_{\epsilon_1}) \geq f(\mathcal{X}_{\epsilon_2}) \geq \dots \quad (14)$$

or

$$f(\mathcal{X}_{\epsilon_0}) \leq f(\mathcal{X}_{\epsilon_1}) \leq f(\mathcal{X}_{\epsilon_2}) \leq \dots \quad (15)$$

The number of connected components (zeroth Betti number  $\beta_0$ ) and the number of cycles (first Betti number  $\beta_1$ ) satisfy the monotonicity (Figures 13 and 15). The size of the largest cluster also satisfies a similar but opposite relation of monotonic increase. There are numerous monotone graph theory features (Chung et al., 2015a, 2017b).

For graphs,  $\beta_1$  can be computed easily as a function of  $\beta_0$ . Note that the Euler characteristic  $\chi$  can be computed in two different ways

$$\begin{aligned} \chi &= \beta_0 - \beta_1 + \beta_2 - \dots \\ &= \#nodes - \#edges + \#faces - \dots, \end{aligned}$$

where  $\#nodes, \#edges, \#faces$  are the number of nodes, edges and faces. However, graphs do not have filled faces and Betti numbers higher than  $\beta_0$  and  $\beta_1$  can be ignored. Thus, a graph with  $p$  nodes and  $q$  edges is given by (Adler et al., 2010)

$$\chi = \beta_0 - \beta_1 = p - q.$$

Thus,

$$\beta_1 = p - q - \beta_0.$$

In a graph, Betti numbers  $\beta_0$  and  $\beta_1$  are monotone over filtration on edge weights (Chung et al., 2019a,b). When we do filtration

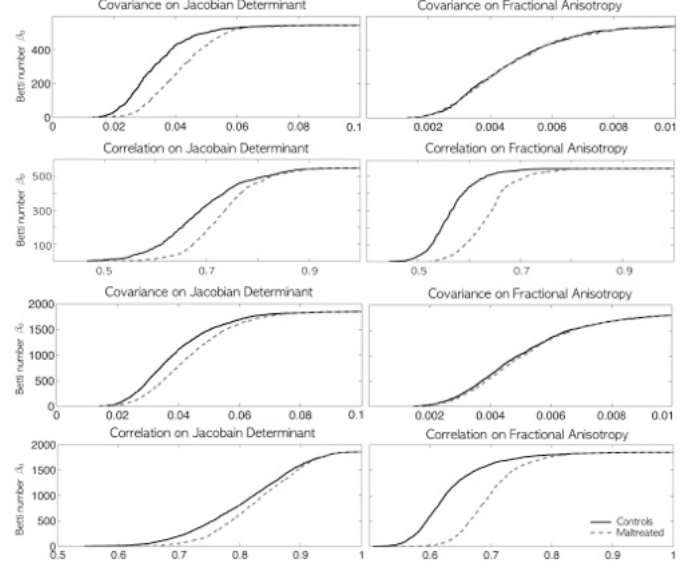


Figure 15: The Betti curves on the covariance correlation matrices for Jacobian determinant (left column) and fractional anisotropy (right column) on 548 (top two rows) and 1856 (bottom two rows) nodes (Chung et al., 2015a). Unlike the covariance, the correlation seems to show huge group separation between normal and maltreated children visually. However, in all 7 cases except top right (548 nodes covariance for FA), statistically significant differences were detected using the rank-sum test on the areas under the Betti-plots ( $p$ -value  $< 0.001$ ). The shapes of Betti-plots are consistent between the studies with different node sizes indicating the robustness of the proposed method over changing number of nodes.

on the maximal filtration in (14), edges are deleted one at a time. Since an edge has only two end points, the deletion of an edge disconnect the graph into at most two. Thus, the number of connected components ( $\beta_0$ ) always increases and the increase is at most by one. Note  $p$  is fixed over the filtration but  $q$  is decreasing by one while  $\beta_0$  increases at most by one. Hence,  $\beta_1$  always decreases and the decrease is at most by one. Further, the length of the largest cycles, as measured by the number of nodes, also decreases monotonically (Figure 16).

Identifying connected components in a network is important to understand in decomposing the network into disjoint subnetworks. The number of connected components (0-th Betti number) of a graph is a topological invariant that measures the number of structurally independent or disjoint subnetworks. There are many available existing algorithms, which are not related to persistent homology, for computing the number of connected components including the Dulmage-Mendelsohn decomposition (Pothen and Fan, 1990), which has been widely used for decomposing sparse matrices into block triangular forms in speeding up matrix operations.

In graph filtrations, the number of cycles increase or decreases as the filtration value increases. The pattern of monotone increase or decrease can visually show how the topology of the graph changes over filtration values. The overall pattern of *Betti curves* can be used as a summary measure of quantifying how the graph changes over increasing edge weights (Chung et al., 2013) (Figure 13). The Betti curves are related to barcodes. The Betti number is equal to the number of bars in the



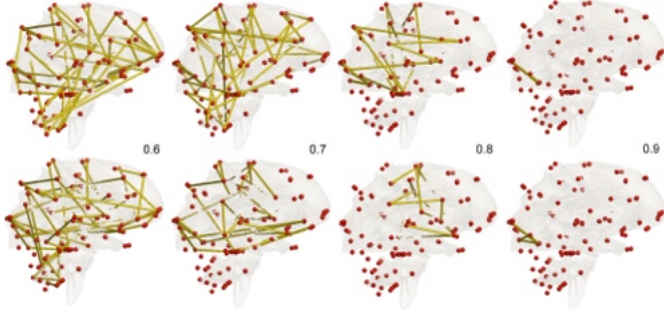


Figure 16: The largest cycle at given correlation thresholds on rs-fMRI. Two representative subjects in HCP were used (Chung et al., 2019a). As the threshold increases, the length of cycles decreases monotonically.

barcodes at the specific filtration value.

### 6.3. Graph filtration in trees

Binary trees have been a popular data structure to analyze using persistent homology in recent years (Bendich et al., 2016; Li et al., 2017). Trees and graphs are 1-skeletons, which are Rips complexes consisting of only nodes and edges. However, trees do not have 1-cycles and can be quantified using up to 0-cycles only, i.e., connected components, and higher order topological features can be simply ignored. However, Garside et al. (2020) used somewhat inefficient filtrations in the 2D plane that increase the radius of circles from the root node or points along the circles. Such filtrations will produce persistent diagrams (PD) that spread points in 2D plane. Further, it may create 1-cycles. Such PD are difficult to analyze since scatter points do not correspond across different PD. For 1-skeleton, the *graph filtration* offers more efficient alternative (Chung et al., 2019b; Songdechakraiut and Chung, 2020b).

Consider a tree  $\mathcal{T} = (V, w)$  with node set  $V = \{1, 2, \dots, p\}$  and weighted adjacency matrix  $w$ . If we have binary tree with binary adjacency matrix, we add edge weights by taking the distance between nodes  $i$  and  $j$  as the edge weights  $w_{ij}$  and build a weighted tree with  $w = (w_{ij})$ . For a tree  $T$  with  $p \geq 2$  nodes and unique  $p - 1$  positive edge weights  $w_{(1)} < w_{(2)} < \dots < w_{(p-1)}$ . Threshold  $\mathcal{T}$  at  $\epsilon$  and define the binary tree  $\mathcal{T}_\epsilon = (V, w_\epsilon)$  with edge weights  $w_\epsilon = (w_{\epsilon,ij})$ ,  $w_{\epsilon,ij} = 1$  if  $w_{ij} > \epsilon$  and 0 otherwise. Then we have graph filtration on trees

$$\mathcal{T}_{w_{(0)}} \supset \mathcal{T}_{w_{(1)}} \supset \mathcal{T}_{w_{(2)}} \supset \dots \supset \mathcal{T}_{w_{(p-1)}}. \quad (16)$$

Since all the edge weights are above filtration value  $w_{(0)} = 0$ , all the nodes are connected, i.e.,  $\beta_0(w_{(0)}) = 1$ . Since no edge weight is above the threshold  $w_{(p-1)}$ ,  $\beta_0(w_{(p-1)}) = p$ . Each time we threshold, the tree splits into two and the number of components increases exactly by one in the tree (Chung et al., 2019b). Thus, we have

$$\beta_0(\mathcal{T}_{w_{(1)}}) = 2, \beta_0(\mathcal{T}_{w_{(2)}}) = 3, \dots, \beta_0(\mathcal{T}_{w_{(p-1)}}) = p.$$

Thus, the coordinates for the 0-th Betti curve is given by

$$(0, 1), (w_{(1)}, 2), \dots, (w_{(2)}, 3), (w_{(p-1)}, p), (\infty, p).$$

All the 0-cycles (connected components) never die once they are born over graph filtration. For convenience, we simply let the death value of 0-cycles at some fixed number  $c > w_{(q-1)}$ . Then PD of the graph filtration is simply

$$(w_{(1)}, c), (w_{(2)}, c), \dots, (w_{(q-1)}, c)$$

forming 1D scatter points along the horizontal line  $y = c$  making various operations and analysis on PD significantly simplified (Songdechakraiut and Chung, 2020b). Figure 17 illustrates the graph filtration and corresponding 1D scatter points in PD on the binary tree used in Garside et al. (2020). A different graph filtration is also possible by making the edge weight to be the shortest distance from the root node. However, they should carry the identical topological information.

For a general graph, it is not possible to analytically determine the coordinates for its Betti curves. The best we can do is to compute the number of connected components  $\beta_0$  numerically using the single linkage dendrogram method (SLD) (Lee et al., 2012), the Dulmage-Mendelsohn decomposition (Pothén and Fan, 1990; Chung et al., 2011) or through the Gaussian elimination (de Silva and Ghrist, 2007; Carlsson and Memoli, 2008; Edelsbrunner et al., 2002).

### 6.4. Node-based filtration

Instead of doing graph filtration at the edge level, it is possible to build different kind of filtrations at the node level (Hofer et al., 2020; Wang et al., 2017b). Consider graph  $G = (V, E)$  with nodes  $V = 1, 2, \dots, p$  and node weights  $w_i$  defined at each node  $i$ . With threshold  $\lambda$ , define a binary network  $G_\lambda = (V_\lambda, E_\lambda)$  where

$$V_\lambda = \{i \in V : w_i \leq \lambda\}$$

and

$$E_\lambda = \{(i, j) \in E : \max(w_i, w_j) \leq \lambda\}.$$

Note  $E_\lambda \subset E$  such that two nodes  $i$  and  $j$  are connected if  $\max(w_i, w_j) \leq \lambda$ . We include a node from  $G$  in  $G_\lambda$  when the threshold  $\lambda$  is above its weight, and we connect two nodes in  $G_\lambda$  with an edge when  $\lambda$  is above the larger weight of any of the two nodes. Then we have the *node-based* graph filtration

$$G_{w_{(1)}} \subset G_{w_{(2)}} \subset \dots \subset G_{w_{(q)}}. \quad (17)$$

The filtration (17) is not affected by reindexing nodes since the edge weights remain the same regardless of node indexing. Each  $G_{w_{(j)}}$  in (17) consists of clusters of connected nodes; as  $\lambda$  increases, clusters appear and later merge with existing clusters. The pattern of changing clusters in (17) has the following properties.

For  $w_{(i)} \leq \lambda < w_{(i+1)}$ ,  $G_\lambda = G_{w_{(i)}}$ , the filtration (17) is maximal in the sense that no more  $G_\lambda$  can be added to (17). As  $\lambda$  increases from  $w_{(i)}$  to  $w_{(i+1)}$ , only the node  $v'_{i+1}$  that corresponds to the weight  $w_{(i+1)}$  is added in  $V_{w_{(i+1)}}$ .

Node-based graph filtration was applied to the Delaunay triangulation of EEG channels in the meditation study in Wang et al. (2017b). Node weights are the powers at the EEG channels. At each filtration value  $\lambda$ , both the nodes and edges with

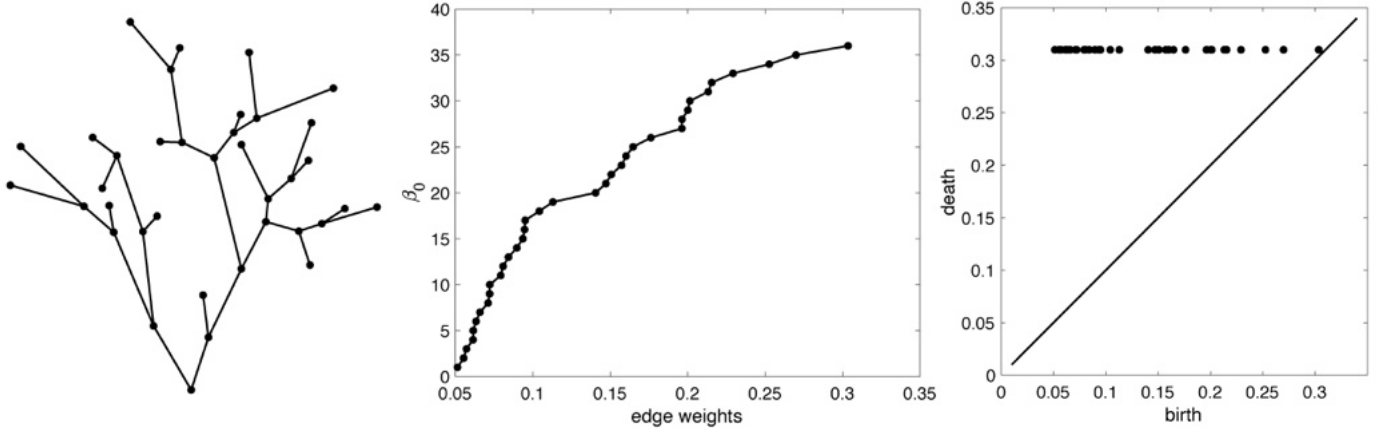


Figure 17: Left: binary tree used in [Garside et al., 2020](#). Middle:  $\beta_0$ -curve over graph filtration. Edge weights of the tree is used as the filtration values. Right: The points in the persistent diagram all lined up at  $y = 0.31$ , which is arbitrarily picked to be larger than the maximum edge weight 0.3034.

weights less than or equal to  $\lambda$  are added. The clusters change as  $\lambda$  increases.

Over the years, other filtrations on graph and networks have been developed including clique filtration ([Petri et al., 2013](#); [Stolz et al., 2017](#)).

## 7. Topological distances

The topological distances and losses are usually built on top of various algebraic representations of persistent homology such as barcodes, PD and graph filtrations. The Gromov-Hausdorff (GH) distance is possibly the most popular distance that is originally used to measure distance between two metric spaces ([Tuzhilin, 2016](#)). It was later adapted to measure distances in persistent homology, dendrograms ([Carlsson and Memoli, 2008](#); [Carlsson and Mémoli, 2010](#); [Chazal et al., 2009](#)) and brain networks ([Lee et al., 2011a, 2012](#)). The probability distributions of GH-distance is unknown. Thus, the statistical inference on GH-distance has been done through resampling techniques such as jackknife, bootstraps or permutations ([Lee et al., 2012, 2017](#); [Chung et al., 2015a](#)), which often cause computational bottlenecks for large-scale networks. To bypass the computational bottleneck associated with resampling large-scale networks, the *Kolmogorov-Smirnov (KS) distance* was introduced in ([Chung, 2012](#); [Chung et al., 2017b](#); [Lee et al., 2017](#)). In this section, we review various topological distances.

### 7.1. Bottleneck distance

This is perhaps the most often used distance in persistent homology but it is rarely useful in applications due to the crude nature of metric. Given two networks  $\mathcal{X}^1 = (V^1, w^1)$  with  $m$  cycles and  $\mathcal{X}^2 = (V^2, w^2)$  with  $n$  cycles, we construct a filtration. Subsequently, PDs

$$\mathcal{P}(\mathcal{X}^1) = \{(\xi_1^1, \tau_1^1), \dots, (\xi_m^1, \tau_m^1)\}$$

and

$$\mathcal{P}(\mathcal{X}^2) = \{(\xi_1^2, \tau_1^2), \dots, (\xi_n^2, \tau_n^2)\}$$

are obtained through the filtration ([Lee et al., 2012](#); [Chung et al., 2019b](#)). The bottleneck distance between the networks is defined as the bottleneck distance of the corresponding PDs and bound the Hausdorff distance ([Cohen-Steiner et al., 2007](#); [Edelsbrunner and Harer, 2008](#)):

$$D_B(\mathcal{P}(\mathcal{X}^1), \mathcal{P}(\mathcal{X}^2)) = \inf_{\gamma} \sup_{1 \leq i \leq m} \|t_i^1 - \gamma(t_i^1)\|_{\infty}, \quad (18)$$

where  $t_i^1 = (\xi_i^1, \tau_i^1) \in \mathcal{P}(\mathcal{X}^1)$  and  $\gamma$  is a bijection from  $\mathcal{P}(\mathcal{X}^1)$  to  $\mathcal{P}(\mathcal{X}^2)$ . The infimum is taken over all possible bijections. If  $t_j^2 = (\xi_j^2, \tau_j^2) = \gamma(t_i^1)$  for some  $i$  and  $j$ ,  $L_{\infty}$ -norm is given by

$$\|t_i^1 - \gamma(t_i^1)\|_{\infty} = \max(|\xi_i^1 - \xi_j^2|, |\tau_i^1 - \tau_j^2|).$$

The optimal bijection  $\gamma$  is often determined by the Hungarian algorithm ([Cohen-Steiner et al., 2007](#); [Edelsbrunner and Harer, 2008](#)). Note [\(18\)](#) assumes  $m = n$  such that the bijection  $\gamma$  exists. If  $m \neq n$ , there is no one-to-one correspondence between two PDs. Then, additional points should be augmented along the diagonal line to match unmatched pairs. This enables us to match short-lived homology classes to zero persistence ([Chung et al., 2019b](#); [Cole, 2020](#)).

If the two networks share the same node set  $V^1 = V^2$ , with  $p$  nodes and the same number of  $q$  unique edge weights. If the graph filtration is performed on two networks, the number of their 0D and 1D cycles that appear and disappear during the filtration is  $p$  and  $1 - p + q$ , respectively ([Chung et al., 2019b](#)). Thus, their persistence diagrams of 0D and 1D cycles always have the same number of points.

The well known stability theorem ([Cohen-Steiner et al., 2007](#)) states

$$D_B(\mathcal{P}(\mathcal{X}^1), \mathcal{P}(\mathcal{X}^2)) \leq \|w^1 - w^2\|_{\infty}.$$

The stability theorem is established on Morse functions on a compact manifold but should be true for most of applications. Since the infinity norm is a crude metric even in the Euclidean space, such stability theorem does not imply statistical sensitivity or robustness.

## 7.2. Wasserstein distance

The bottleneck distance is not necessarily a sensitive metric and usually performs poorly compared to other distances (Lee et al., 2011a; Chung et al., 2017a). A more sensitive distance might be the  $q$ -Wasserstein distance (Edelsbrunner and Harer 2010b) which is related to recently popular *optimal transports*.  $q$ -Wasserstein distance  $D_W$  is defined as

$$D_W(\mathcal{P}(\mathcal{X}^1), \mathcal{P}(\mathcal{X}^2)) = \left[ \inf_{\gamma} \sum_i \|t_i^1 - \gamma(t_i^1)\|_{\infty}^q \right]^{1/q},$$

where the infimum is taken over all bijections  $\gamma$  between  $\mathcal{P}(\mathcal{X}^1)$  and  $\mathcal{P}(\mathcal{X}^2)$ , with the possibility of augmenting unmatched points to the diagonal (Chung et al., 2019b). It can be shown that the  $q$ -Wasserstein distance is bounded by

$$D_W(\mathcal{P}(\mathcal{X}^1), \mathcal{P}(\mathcal{X}^2)) \leq C \left[ \inf \sum \|f - g\|_{\infty}^q \right]^{1/q}$$

for some constant  $C$ . Again since the infinity norm is a crude metric, the stability statement simply implies the distance are well bounded and behave reasonably well but it does not implies statistical sensitivity.

## 7.3. Gromov-Hausdorff distance

Gromov-Hausdorff (GH) distance for brain networks is introduced in Lee et al. (2011a, 2012). GH-distance measures the difference between networks by embedding the network into the ultrametric space that represents hierarchical clustering structure of network (Carlsson and Mémoli 2010). The distance  $s_{ij}$  between the closest nodes in the two disjoint connected components  $\mathbf{R}_1$  and  $\mathbf{R}_2$  is called the single linkage distance (SLD), which is defined as

$$s_{ij} = \min_{l \in \mathbf{R}_1, k \in \mathbf{R}_2} w_{lk}.$$

Every edge connecting a node in  $\mathbf{R}_1$  to a node in  $\mathbf{R}_2$  has the same SLD. SLD is then used to construct the single linkage matrix (SLM)  $S = (s_{ij})$ . SLM shows how connected components are merged locally and can be used in constructing a dendrogram. SLM is a *ultrametric* which is a metric space satisfying the stronger triangle inequality  $s_{ij} \leq \max(s_{ik}, s_{kj})$  (Carlsson and Mémoli 2010). Thus the dendrogram can be represented as an ultrametric space  $\mathcal{D} = (V, S)$ , which is again a metric space. GH-distance between networks is then defined through GH-distance between corresponding dendrograms. Given two dendrograms  $\mathcal{D}^1 = (V, S^1)$  and  $\mathcal{D}^2 = (V, S^2)$  with SLM  $S^1 = (s_{ij}^1)$  and  $S^2 = (s_{ij}^2)$ ,

$$D_{GH}(\mathcal{D}^1, \mathcal{D}^2) = \max_{i,j} |s_{ij}^1 - s_{ij}^2|. \quad (19)$$

GH-distance is the maximum of single linkage distance (SLD) differences (Lee et al., 2012). SLD between two nodes  $i$  and  $j$  is the shortest distance between two connected components that contain the nodes. The edges that gives the the maximum SLM is the GH-distance between the two networks. Among all topological distances, only GH-distance uses the single linkage clustering. All other distances do not use the single linkage

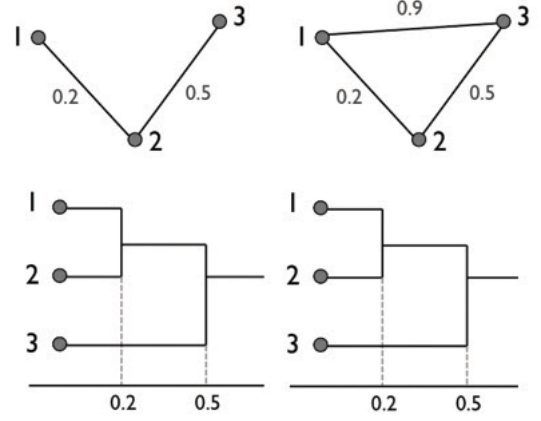


Figure 18: Two topologically distinct graphs may have identical dendrograms, which results in zero GH-distance.

clustering (Chung et al., 2017a). For instance, bottleneck distance or KS-distance are not related to single linkage clustering (Lee et al., 2012).

The limitation of the SLM is the inability to discriminate a cycle in a graph. Consider two topologically different graphs with three nodes (Figure 18). However, the corresponding SLM are identically given by

$$\begin{pmatrix} 0 & 0.2 & 0.5 \\ 0.2 & 0 & 0.5 \\ 0.5 & 0.5 & 0 \end{pmatrix} \text{ and } \begin{pmatrix} 0 & 0.2 & 0.5 \\ 0.2 & 0 & 0.5 \\ 0.5 & 0.5 & 0 \end{pmatrix}.$$

The lack of uniqueness of SLMs makes GH-distance incapable of discriminating networks with cycles (Chung 2012).

For the statistical inference on GH-distance, resampling techniques such as jackknife or permutation tests are often used (Lee et al., 2011a, 2012, 2017).

## 7.4. Kolmogorov-Smirnov distance

Many TDA distances are not scalable and may not be applicable for large brain networks at the voxel level without simplification on the underlying algebraic representations. Motivated by the need to build a scalable topological distance, Kolmogorov-Smirnov (KS) distance was introduced in Chung et al. (2013, 2015a, 2017b); Lee et al. (2017). KS-distance is built on top of the graph filtration, a reduced Rips filtration on 1-skeleton. Given two networks  $\mathcal{X}^1 = (V, w^1)$  and  $\mathcal{X}^2 = (V, w^2)$ , KS-distance between  $\mathcal{X}^1$  and  $\mathcal{X}^2$  is defined as

$$D_{KS}(\mathcal{X}^1, \mathcal{X}^2) = \sup_{\epsilon \geq 0} |\beta_i(\mathcal{X}_\epsilon^1) - \beta_i(\mathcal{X}_\epsilon^2)|$$

for  $\beta_0$  and  $\beta_1$  curves on graph filtrations  $\mathcal{X}_\epsilon^1$  and  $\mathcal{X}_\epsilon^2$ . The distance  $D_{KS}$  is motivated by the KS-test for determining the equivalence of two cumulative distribution functions in non-parametric statistics (Böhm and Hornik, 2010; Chung et al., 2017b; Gibbons and Chakraborti, 2011). The distance  $D_{KS}$  can be discretely approximated using the finite number of filtrations  $\epsilon_1, \dots, \epsilon_q$ :

$$D_q = \sup_{1 \leq j \leq q} |\beta_i(\mathcal{X}_{\epsilon_j}^1) - \beta_i(\mathcal{X}_{\epsilon_j}^2)|.$$

If we choose enough number of  $q$  such that  $\epsilon_j$  are all the sorted edge weights, then (Chung et al., 2017b).

$$D_{KS}(\mathcal{X}^1, \mathcal{X}^2) = D_q.$$

This is possible since there are only up to  $p(p-1)/2$  number of unique edges in a graph with  $p$  nodes and the monotone function increases discretely but *not continuously*. In practice,  $\epsilon_j$  may be chosen uniformly at discrete intervals.

KS-distance treat the two networks in Figure 18 as identical if Betti number  $\beta_0$  is used as the monotonic feature function. To discriminate cycles, KS-distance on  $\beta_1$  should be used. This example illustrates a need for new more sophisticated network distance that can discriminates the higher order topological features beyond connected components. The advantage of using KS-distance is its easiness to interpret compared to other less intuitive distances from persistent homology. Due to its simplicity, it is possible to determine its probability distribution exactly (Chung et al., 2017b).

## 8. Topological inference and learning

It is difficult to directly apply existing statistical and machine learning methods to TDA features such as PDs and barcodes that do not have one-to-one correspondence across features. It is difficult to average such features without transformations or smoothing the features first. Thus, there have been consistent lack of the concept of statistical consistency and statistical methods in the field. One may assume that resampling based statistical inference with minimal statistical assumptions such as the permutation test may work. However, resampling directly on the Rips complex or other derived topological feature may not be feasible for large-scale networks. Thus, there is a strong need to develop a scalable statistical inference procedures tailored for TDA methods (Chung et al., 2017b, 2019b).

### 8.1. Statistical consistency

Most TDA features have the stability results that show the distance between two topological features are bounded by some known distance (Cohen-Steiner et al., 2007; Adams et al., 2017). Such stability results do not necessarily provide the *statistical consistency* that is required for building proper test statistics for hypothesis testing that is needed in brain imaging (Bubenik et al., 2010). The test statistic  $T_n$ , which depends on the sample size  $n$ , is *consistent* if it converge to true value value  $T$  in probability as  $n$  goes to infinity. For  $T_n$  to be consistent, we need

$$\lim_{n \rightarrow \infty} P(|T_n - T| > \epsilon) = 0$$

for all  $\epsilon > 0$ . Most statistics such as sample mean and  $T$ -statistics are all consistent. The consistency grantees the convergence of statistical results when enough samples are used. Existing stability results are mostly on the stability of TDA features but not about the stability or consistency of the test statistics on such features. So additional investigations are needed to establish the consistency of statistics built on top of TDA features, which were rarely done till now.

### 8.2. Exact topological inference

The easiest way to build consistent statistical procedures in TDA is using the available topological distances with well established stability results. Unfortunately, there is no known probability distributions for topological distances except KS-distance (Chung et al., 2019b). The complicated algebraic forms of most TDA distances do not make building probability distributions on them easier. The probability distribution of KS-distance  $D_q$  under the null assumption of the equivalence of two Betti curves  $\beta_i(\mathcal{X}_{\epsilon_j}^1)$  and  $\beta_i(\mathcal{X}_{\epsilon_j}^2)$  is asymptotically given by (Chung et al., 2017b, 2019b)

$$\lim_{q \rightarrow \infty} (D_q / \sqrt{2q} \geq d) = 2 \sum_{i=1}^{\infty} (-1)^{i-1} e^{-2i^2 d^2}. \quad (20)$$

The  $p$ -value under the null is then computed as

$$p\text{-value} = 2e^{-d_o^2} - 2e^{-8d_o^2} + 2e^{-18d_o^2} \dots,$$

where the observed value  $d_o$  is the least integer greater than  $D_q / \sqrt{2q}$  in the data. For any large value  $d_o \geq 2$ , the second term is in the order of  $10^{-14}$  and insignificant. Even for small observed  $d_o$ , the expansion converges quickly. The KS-distance does not assume any statistical distribution. This is perhaps the only known distribution related to topological distances. The main advantage of the method is that the method avoid using time the consuming permutation test for large-scale networks.

For other distances, resampling techniques such as permutations and bootstraps are needed to determine the statistical distributions.

### 8.3. Permutation test

The permutation test (Fisher 1966) is the most widely used nonparametric test procedure in brain imaging, which can be easily adapted for inference on distance and loss functions. It is known as the only *exact test* since the distribution of the test statistic under the null hypothesis can be exactly computed by calculating all possible values of the test statistic under every possible combinations. Unfortunately, even for modest sample sizes, the total number of permutations is astronomically large and only a small subset of permutations is used in approximating  $p$ -values. Thus, permutation tests are all approximate in practice.

The permutation test for two samples are done as follows (Chung et al., 2013; Efron 1982; Lee et al., 2012). Two sample test setting is probably the most often encountered in brain networks. Suppose there are  $m$  networks  $\mathcal{X}_1, \mathcal{X}_2, \dots, \mathcal{X}_m$  in Group I and  $n$  networks  $\mathcal{Y}_1, \mathcal{Y}_2, \dots, \mathcal{Y}_n$  in Group II. We are interested in testing if networks in two groups differ. Concatenate all the networks and reindex them as

$$\mathcal{Z} = (\mathcal{Z}_1, \dots, \mathcal{Z}_{m+n}) = (\mathcal{X}_1, \dots, \mathcal{X}_m, \mathcal{Y}_1, \dots, \mathcal{Y}_n).$$

Following Songdechakrai and Chung (2020b), we define the average between-group topological distance as

$$D_B(\mathcal{Z}) = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n D(\mathcal{X}_i, \mathcal{Y}_j).$$



Any topological distance  $D$  can be used for this purpose. The smaller distance  $D_B$  implies smaller group differences. Thus,  $D_B$  can be used as the test statistic for differentiating the groups. For this, we need to determine the empirical distribution of  $D_B$  under the null hypothesis of no group difference. Under the null, the group labels are interchangeable and the sample space can be generated by permutations. The permutations are done as follows. Now permute the indices of  $\mathcal{Z}$  in the symmetric group of degree  $m + n$ , i.e.,  $S_{m+n}$  (Kondor et al., 2007). The permuted networks are indexed as  $\mathcal{Z}_\sigma$ , where  $\sigma \in S_{m+n}$  is the permutation. Then we split  $\mathcal{Z}_\sigma$  into two parts

$$\mathcal{Z}_\sigma = (\mathcal{Z}_{\sigma(1)}, \dots, \mathcal{Z}_{\sigma(m)}, \mathcal{Z}_{\sigma(m+1)}, \dots, \mathcal{Z}_{\sigma(m+n)}).$$

Then for each permutation  $\sigma$ , we compute  $D_B(\mathcal{Z}_\sigma)$ . Theoretically, for all  $S_{m+n}$ , we can compute the distances. This gives the empirical estimation of the distribution of  $D_B$ . The number of permutations exponentially increases and it is impractical to generate every possible permutation. So up to tens of thousands permutations are usually generated to approximate the null distribution. This is an approximate method and a care should be taken to guarantee the convergence. For faster convergence, we can use the transposition test, which iteratively computes the next permutation from the previous permutation in a sequential manner (Chung et al., 2019c; Songdechakraiut and Chung, 2020b).

#### 8.4. Learning across brain networks

Many existing prediction and classification models of brain structures and functions are usually based on a specific brain parcellation scheme. Summary measurements across parcellations are compared through various standard loss functions such as correlations (Huang et al., 2020b) and mutual information (Thirion et al., 2014). However, there begin to emerge evidences that brain functions do not correlate well with the fixed parcellation boundaries (Rottschy et al., 2012; Eickhoff et al., 2016; Arslan et al., 2018; Kong et al., 2019). Many existing parcellation schemes give raise to conflicting topological structures over different scales (Lee et al., 2012; Chung et al., 2015a, 2018). The topological structure of network at one parcellation scale may not carry over to different parcellation scales. For instance, the estimated modular structure usually do not have continuity over different resolution parameters or parcellations (Betzel and Bassett, 2017; Chung, 2018). Thus, there are needs to develop learning frameworks that provide a consistent and complete picture of brain anatomical structure and functional processes regardless of the choice of parcellation. One possible solution is to develop learning framework across parcellations of different sizes and topology.

Existing prediction models mainly use regressions, such as general linear models (GLM), logistic regressions or mixed-effect models, that incorporate the accumulated effect of features as the sum of predictive variables in correlating cognitive scores (Zhang et al., 2019). Regression models might be reasonable for determining the group level average patterns. However, the underlying network features that matter most and in what combinations might be just too complex to be discovered

in the regression based predictive models. These challenges can be addressed through *deep learning*, which is methodologically well suited to address all these challenges.

Due to the popularity of deep learning, there have been recent attempts at trying to incorporate the persistent homology to deep learning (Chen et al., 2019; Hu et al., 2019). Earlier attempts at incorporating topology into learning framework was not effective since they mainly used topological features into classification frameworks (Guo et al., 2018; Garin and Tauzin, 2019). Instead of trying to feed the scalar summary of topological features that may not be effective, more efficient approaches seem to use loss and cost functions that explicitly incorporate topology in learning.

#### 8.5. Topological loss

For various learning tasks such as clustering, classification and regression, traditional losses mostly based on Euclidean distance are often used (Chung et al., 2019b). Such loss functions are well suited for measuring similarity and distance between homogenous imaging data that match across subjects. For heterogenous data such as brain sulcal patterns (Huang et al., 2020a) or functional activation patterns that may not have voxel-to-voxel correspondence across subjects (Desai et al., 2005), such loss functions are ill-suited or ineffective. Also it is not straightforward to set up loss function between brain parcellations of different sizes and topology. However, the topological distances we reviewed can easily handle such heterogenous data of different sizes and topology.

In addition to topological distances we reviewed so far, we introduce an additional topological loss function in terms of the barcodes (Edelsbrunner and Harer, 2010a; Songdechakraiut and Chung, 2020a). The expensive optimization process involved in the Rips filtration takes  $O(d^6)$  run-time for  $d$  number of data points (Edmonds and Karp, 1972; Kerber et al., 2017; Edelsbrunner and Harer, 2010a), making it impractical in brain networks with far larger number of topological features involving hundreds of connected components and thousands of cycles. Instead of the Rips filtration, if we use the graph filtration, the optimization problem turns into an optimal matching problem with a significantly reduced run-time of  $O(d^2 \log d)$  (Songdechakraiut and Chung, 2020a).

Consider a network  $G = (V, w)$  comprising a set of nodes  $V$  and unique positive symmetric edge weights  $w = (w_{ij})$ . The graph filtration is defined on  $G$ . Unlike the Rips complex, there are no more higher dimensional topological features to compute in a graph filtration beyond the 1D topology. The 0D or 1D barcode corresponding to the network  $G$  is a collection of intervals  $[b_i, d_i]$  such that each interval tabulates the life-time of a connected component or a cycle that appears at the first filtration value  $b_i$  and vanishes at the last filtration value  $d_i$ .

The number of connected components  $\beta_0$  is non-decreasing as  $\epsilon$  increases, and so we can simply represent the 0D barcode of the network using only the set of increasing birth values:

$$I_0(G) : b_0 < b_1 < \dots < b_{m_0}$$

Similarly the number of 1-cycles is monotonically decreasing, and thus we can represent the 1D barcode using only the set of

increasing death values:

$$I_1(G) : d_0 < d_1 < \dots < d_{m_1}$$

Increasing the filtration  $\epsilon$  can result in either the birth of a connected component or the death of a cycle but not both at the same time. Therefore, the set of 0D birth values  $I_0(G)$  and 1D death values  $I_1(G)$  partition the edge weight set  $W$  such that  $W = I_0(G) \cup I_1(G)$  with  $I_0(G) \cap I_1(G) = \emptyset$  (Songdechakraiwt and Chung, 2020a). This decomposition can be effectively used in computing the topological loss efficiently.

A topological loss function which measures the topological similarity between two networks can be defined in terms of the 0D and 1D barcodes. Let  $G_1 = (V, w^1)$  and  $G_2 = (V, w^2)$  be networks of same size. The topological loss  $L_{top}(G_1, G_2)$  is defined by the optimal matching cost:

$$L_{top}(G_1, G_2) = \min_{\tau} \sum_{b \in I_0(G_1)} [b - \tau(b)]^2 + \sum_{d \in I_1(G_1)} [d - \tau(d)]^2$$

assuming bijection  $\tau : I_0(G_1) \cup I_1(G_1) \rightarrow I_0(G_2) \cup I_1(G_2)$  factorizes  $\tau = \tau_0 \oplus \tau_1$  into bijections  $\tau_0 : I_0(G_1) \rightarrow I_0(G_2)$  and  $\tau_1 : I_1(G_1) \rightarrow I_1(G_2)$ .  $L_{top}$  is the modified Wasserstein distance between the barcodes of the two networks. It can be shown that the optimal bijection  $\tau_0$  is given by matching the  $i$ -th smallest birth values in  $I_0(G_1)$  to the  $i$ -th smallest birth values in  $I_0(G_2)$  (Songdechakraiwt and Chung, 2020a). Similarly  $\tau_1$  is given by matching the  $i$ -th smallest death values in  $I_1(G_1)$  to the  $i$ -th smallest death values in  $I_1(G_2)$ . For two networks of different sizes, any unmatched 0D birth or 1D death values in the larger network are matched to the largest or smallest edge weights in the smaller network, respectively.

A topological loss function is given by the Wasserstein distance between the persistence images of two datasets. Other topological loss is also possible. In (Brüel-Gabrielsson et al., 2019), topological loss  $L_{PD_k}$  on the persistent diagram of  $k$ -cycle was introduced as

$$L_{PD_k}(p, q, i_0) = \sum_{i \geq i_0} (d_i - b_i)^p \left( \frac{d_i + b_i}{2} \right)^q,$$

where  $(b_i, d_i)$  is the  $i$ -th barcode and  $i_0$  is the index for the  $i_0$ -th most persistent point. The power  $p$  can be increased to more strongly penalize the most persistent features, and the power  $q$  serves to weight features that are prominent later in the filtration. We can use  $L_{PD_k}$  as a regularizer that penalizes on the number of clusters or cycles in a natural way that is difficult to accomplish with more traditional analysis.  $L_{PD_k}$  was used in denoising the number of connected component of MNIST image dataset (Brüel-Gabrielsson et al., 2019). Topological regularizers of this type enable us to use the stochastic gradient decent to encourage specific topological features in deep learning tasks.

## 8.6. Topological learning

An important application of deep learning toward brain networks is to uncover patterns in networks, as these patterns can reveal the underlying topological signal. However, much of the deep learning efforts so far has been limited to using the data

structure provided by the *local* geometry. For example, the various Euclidean loss functions often used in generative models as well as for regularizations are defined with only knowledge of the data's local geometry. In contrast, topology encodes the *global* structure of the geometry of data. The purpose of topological learning toward brain networks is to develop learning frameworks that exploit the topological information of networks explicitly.

Central to learning tasks such as regression, classification and clustering is the optimization of a loss function. Given observed data  $X_i$  with responses  $y_i$ , we fit a predictive model with parameters  $\hat{\beta}$  which will enable us to make a prediction  $\hat{y}_i = f(\hat{\beta}; X_i)$  for each observation. The loss function  $L$  assesses the goodness of the fit of the model prediction to the observation, mostly through the Euclidean distance or matrix norms such as the sum of squared residuals

$$L(\beta) = \sum_{i=1}^n (y_i - f(\hat{\beta}; X_i))^2.$$

However, such models are prone to over-fitting if there are more unknown parameters than observations. The regularization term or penalty  $P(\beta)$  is introduced to avoid such overfitting:

$$\hat{\beta} = \arg \min_{\beta} L(\beta) + \lambda P(\beta),$$

where  $\lambda$  is a tuning parameter that controls the contribution of the regularization term. Traditionally common regularizers include  $L_1$  and  $L_2$  loss functions, however, topological penalties are recently introduced to penalize or favor specific topological features (Chen et al., 2019). In (Songdechakraiwt and Chung, 2020b), functional brain networks  $G_1, \dots, G_k$  are regressed in the model

$$\hat{\Theta} = \arg \min_{\Theta} \frac{1}{n} \sum_{k=1}^n L_F(\Theta, G_k) + \lambda L_{top}(\Theta, T),$$

where the Frobenius norm  $L_F$  measures the goodness-of-fit between the network model  $\Theta$  and  $P = (V_P, w_P)$  is a network expressing a prior topological knowledge such as the structural brain network, where functional networks are overlaid. For sufficiently large  $\lambda$ , all the functional networks will be warped to the structural network  $T$  and there will be no topological differences among functional networks.

Topological approaches are relatively unexplored in deep learning. Many past works on incorporating topological features into convolutional neural networks does it implicitly and fail to identify which topological features are learned (Clough et al., 2019). Recently, more explicit topological losses constructed from persistent homology have seen more success in explicitly learning underlying topology of data. Applications of topology to deep learning include the extraction of topological features for better learning (Pun et al., 2018; Cole et al., 2020), using differentiable properties of persistence to define topological loss (Clough et al., 2019; Edelsbrunner and Harer, 2010a; Songdechakraiwt and Chung, 2020a) and persistence images (Cole et al., 2020), deep learning interpretability (Gabrielsson and Carlsson, 2019; Gabella, 2019).

In other applications, we can incorporate *topological priors* in the model or training data to improve the performance in learning tasks (Chen et al., 2019; Br  l-Gabrielsson et al., 2019). In image segmentation, standard loss functions is often constructed at the voxel level ignoring the higher-level structures (Clough et al., 2019). However, it may be necessary to enforce global topological prior to avoid the additional image processing to correct topological defects in the segmentation results (Chung et al., 2015b).

Topology is also used in designing more robust deep learning models Br  l-Gabrielsson et al. (2019). Adversarial attacks in machine learning try to deceive models by supplying deceptive input or outlying data (Huang et al., 2017). Machine learning techniques mainly assume the the training and test data are generated from the statistical distributions. By designing the input data that violates such statistical assumptions, we can fool the learning model. Neutral networks are vulnerable to adversarial attacks which may be imperceptible to the human eye, but can lead the model to misclassify the output (Chakraborty et al., 2018). It has been speculated that models based on topological features are more robust against adversarial attacks (Gabrielsson and Carlsson, 2019; Carlsson and Gabrielsson, 2018). A topological layer can also be placed in neural networks to prevent from adversarial attacks and enhance topological fidelity. In Br  l-Gabrielsson et al. (2019), it was shown that the success rate of adversarial attacks is much smaller on networks with a topological layer. Even though the concept of adversarial attacks have not been popular in brain network modeling, building deep learning models on brain networks that are robust against adversarial attacks would be highly useful.

## 9. Conclusion & Disussion

In this review paper, we surveyed various topological distance and loss functions based on persistent homology. We showed how such distance and loss functions can be used in wide variety of learning applications including regression, clustering and classification. ManyTDA approaches can successfully differentiate the topological network differences. However, often it is unclear where the difference is localized within the networks. In diagnostics related problems such as determining if a subject belong to particular clinical population, it is important to determine the topological differences; however, more important questions is localizing the source of differences. Since there is no one-to-one correspondence between the topological features and the original data, it is often not possible to localize the signals, which has been the biggest limitation of the TDA methods in brain imaging applications. The future development TDA should be toward this important but very difficult question.

Compared to TDA methods, geometric methods are more adapt at detecting localized signals in brain imaging. Figure 19 displays the sulcal and gyral trees obtained from the brain surface mesh (Huang et al., 2019b), where trees are teated as heat source with value +1 on gyral trees and heat sink with value -1 on sulcal trees. Then Isotropic diffusion is applied to produce the smooth map of sulcal and gyral trees. Such smooth

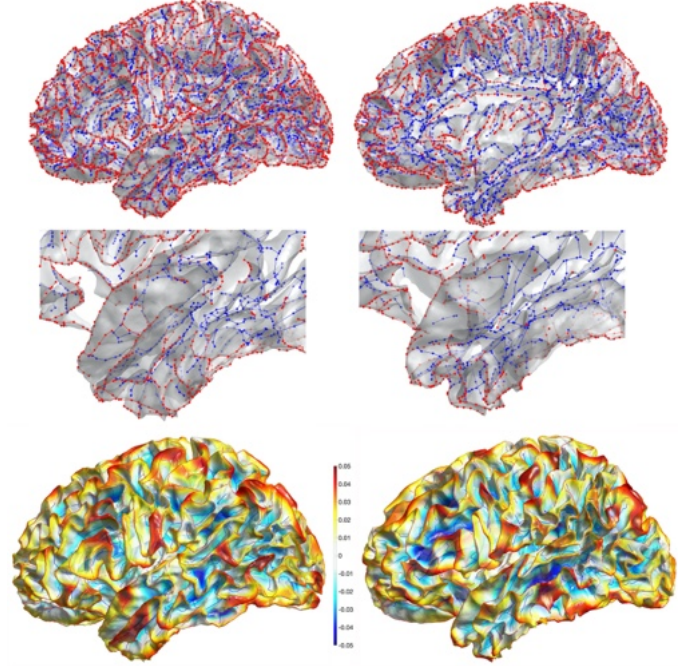


Figure 19: Top: Sulcal (blue) and gyral (red) pattern of two different subjects. The sulcal pattern can be represented as a forest, a collection of trees (Huang et al., 2020a). Since the nodes of the forests are the surface mesh vertices, the forest can be represented as about  $300000 \times 300000$  sparse adjacency matrices. Computing similarity measures between the sulcal forest is not trivial and may require topological distances. Middle: enlargement of the temporal lobe. Bottom: The weighted spherical harmonic representation with degree 70 and heat kernel bandwidth 0.001 (Chung et al., 2008) smoothing the node value where sulci are assigned value -1 and gyri are assigned value 1. The smoothing obviates the need for matching heterogeneous data structure and comparisons of the different sulcal pattern across subject can be done at the mesh vertex level. Such local geometric solution is not readily available for brain networks.

maps can be easily compared across different subjects. For instance, two-sample *t*-statistic at each mesh vertex is performed localizing group differences. Such localized signal detection is difficult if not impossible with many existing TDA methods.

Persistent homology features are by definition global summary measures and they might be more useful for tasks that do not involves identifying the source of signal differences. It might be more useful in discrete decision making tasks such as clustering and classifications. In fact TDA has begin to be more useful in deep learning (Chen et al., 2019).

The paper reviewed how TDA can be used in statistic brain networks that do not change over time. For dynamically changing brain networks over time, it is unclear how TDA can be applied yet. Consider the dynamically changing correlating matrices of resting-state fMRI (Figure 20). At each each correlation matrix, we can apply persistent homology and obtain PD, which is the discrete representation of connectivity at one particular time point. It is unclear how to integrate such discrete topological representation over multiple time points in a continuous fashion. Dynamic-TDA as introduced in Songdechakrui-wut and Chung (2020a) encodes rsfMRI as a time-ordered sequence of Rips complexes and their corresponding barcodes in



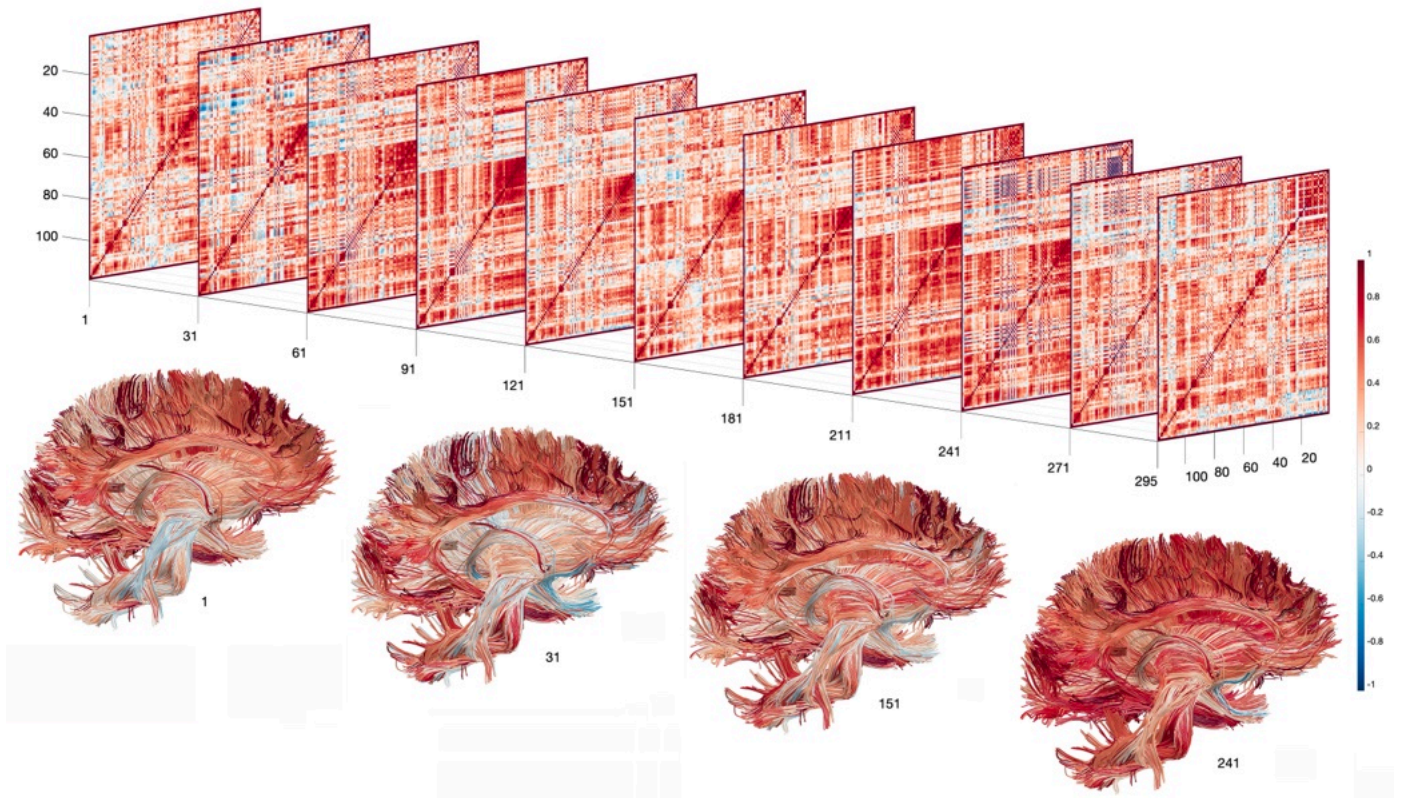


Figure 20: Top: Dynamically changing correlation matrices computed from rs-fMRI using the sliding window of size 60 for a subject (Huang et al., 2019a). Bottom: The constructed correlation matrices are superimposed on top of the white matter fibers, which are colored based on correlations between parcellations. The structural brain network is a more or less a tree like structure without many loops or cycles while the functional brain networks can have many loops and cycles. Thus, aligning functional brain networks on top of structural brain network directly will not work. It requires a new topological approach for aligning topologically different networks.

studying dynamically changing topological patterns over time. Dyanmic-TDA may provide some suggestion for extending TDA to time varying brain networks. This is left as promising future research direction.

## Acknowledgements

This study was supported by NIH grants R01 EB022856 and R01 EB028753 and NSF grant MDS-2010778. We would like to thank Andery Gritsenko of Northeastern University for the computation of the largest cycle in a graph. We also like to thank Alex Cole of University of Amsterdam for the discussion of the witness complex. We thank Hyekyung Lee of Seoul National University Hospital for discussion on sparse filtrations and kernel distances. We would like to thank Robin Henderson of Newcastle University, U.K. for providing the coordinates and connectivity information of nodes of the binary tree used in Garside et al. (2020). We would like to thank Hill Goldsmith of University of Wisconsin for providing one subject rsfMRI data used in the figure, which came from the study Huang et al. (2019a). We thank Taniguchi Masanobu of Waseda University for discussion on canonical correlations.

## References

## References

- Achard, S., Salvador, R., Whitcher, B., Suckling, J., Bullmore, E., 2006. A resilient, low-frequency, small-world human brain functional network with highly connected association cortical hubs. *The Journal of Neuroscience* 26, 63–72.
- Adams, H., Emerson, T., Kirby, M., Neville, R., Peterson, C., Shipman, P., Chepushtanova, S., Hanson, E., Motta, F., Ziegelmeier, L., 2017. Persistence images: A stable vector representation of persistent homology. *The Journal of Machine Learning Research* 18, 218–252.
- Adcock, A., Carlsson, E., Carlsson, G., 2013. The ring of algebraic functions on persistence bar codes. *arXiv preprint arXiv:1304.0530*.
- Adler, R., Bobrowski, O., Borman, M., Subag, E., Weinberger, S., 2010. Persistent homology for random fields and complexes. In: *Borrowing strength: theory powering applications—a Festschrift for Lawrence D. Brown*. Institute of Mathematical Statistics, pp. 124–143.
- Andrew, G., Arora, R., Bilmes, J., Livescu, K., 2013. Deep canonical correlation analysis. In: *International conference on machine learning*. PMLR, pp. 1247–1255.
- Anirudh, R., Thiagarajan, J., Kim, I., Polonik, W., 2016. Autism spectrum disorder classification using graph kernels on multidimensional time series. *arXiv preprint arXiv:1611.09897*.
- Antoine, J., Petra, A., Max, A., 1996. Evaluation of ridge seeking operators for multimodality medical image matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Arafat, N. A., Basu, D., Bressan, S., 2019. Topological data analysis with  $\epsilon$ -net induced lazy witness complex. In: *International Conference on Database and Expert Systems Applications*. Springer, pp. 376–392.



- Arslan, S., Ktena, S., Makropoulos, A., Robinson, E., Rueckert, D., Parisot, S., 2018. Human brain mapping: A systematic comparison of parcellation methods for the human cerebral cortex. *NeuroImage* 170, 5–30.
- Arunkumar, N., Mohammed, M. A., Abd Ghani, M. K., Ibrahim, D. A., Abdulhay, E., Ramirez-Gonzalez, G., de Albuquerque, V. H. C., 2019. K-means clustering and neural network for object detecting and identifying abnormality of brain tumor. *Soft Computing* 23 (19), 9083–9096.
- Ashburner, J., Friston, K., 2000. Voxel-based morphometry - the methods. *NeuroImage* 11, 805–821.
- Asllani, I., Borogovac, A., Brown, T. R., 2008. Regression algorithm correcting for partial volume effects in arterial spin labeling mri. *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 60 (6), 1362–1371.
- Atienza, N., González-Díaz, R., Soriano-Trigueros, M., 2018. On the stability of persistent entropy and new summary functions for tda. *arXiv preprint arXiv:1803.08304*.
- Attali, D., Edelsbrunner, H., Mileyko, Y., 2007. Weak witnesses for delaunay triangulations of submanifolds. In: *Proceedings of the 2007 ACM symposium on Solid and physical modeling*. pp. 143–150.
- Avants, B., Cook, P., Ungar, L., Gee, J., Grossman, M., 2010. Dementia induces correlated reductions in white matter integrity and cortical thickness: a multivariate neuroimaging study with sparse canonical correlation analysis. *NeuroImage* 50, 1004–1016.
- Babai, L., Luks, E., 1983. Canonical labeling of graphs. In: *Proceedings of the fifteenth annual ACM symposium on Theory of computing*. pp. 171–183.
- Banks, D., Carley, K., 1994. Metric inference for social networks. *Journal of classification* 11, 121–149.
- Bassett, D., 2006. Small-world brain networks. *The Neuroscientist* 12, 512–523.
- Bassett, D. S., Wymbs, N. F., Porter, M. A., Mucha, P. J., Grafton, S. T., 2014. Cross-linked structure of network evolution. *Chaos: An Interdisciplinary Journal of Nonlinear Science* 24 (1), 013112.
- Bauer, U., Lesnick, M., 2013. Induced matchings and the algebraic stability of persistence barcodes. *arXiv preprint arXiv:1311.3681*.
- Bendich, P., Marron, J., Miller, E., Pieloch, A., Skwerer, S., 2016. Persistent homology analysis of brain artery trees. *The annals of applied statistics* 10, 198.
- Betzal, R., Bassett, D., 2017. Multi-scale brain networks. *NeuroImage in press*.
- Biagetti, M., Cole, A., Shiu, G., 9 2020. The Persistence of Large Scale Structures I: Primordial non-Gaussianity.
- Bien, J., Tibshirani, R., 2011. Hierarchical clustering with prototypes via mini-max linkage. *Journal of American Statistical Association* 106, 1075–1084.
- Biscio, C. A., Møller, J., 2019. The accumulated persistence function, a new useful functional summary statistic for topological data analysis, with a view to brain artery trees and spatial point process applications. *Journal of Computational and Graphical Statistics* 28 (3), 671–681.
- Bishop, C. M., 2006. *Pattern recognition and machine learning*. springer.
- Blessy, S., Sulochana, C. H., 2015. Performance analysis of unsupervised optimal fuzzy clustering algorithm for mri brain tumor segmentation. *Technology and Health Care* 23 (1), 23–35.
- Blumberg, A. J., Gal, I., Mandell, M. A., Pancia, M., 2014. Robust statistics, hypothesis testing, and confidence intervals for persistent homology on metric measure spaces. *Foundations of Computational Mathematics* 14 (4), 745–789.
- Böhm, W., Hornik, K., 2010. A Kolmogorov-Smirnov test for r samples. *Institute for Statistics and Mathematics Research Report Series, Report 105*.
- Boissonnat, J.-D., Devillers, O., Hornus, S., 2009a. Incremental construction of the delaunay triangulation and the delaunay graph in medium dimension. In: *Proceedings of the twenty-fifth annual symposium on Computational geometry*. pp. 208–216.
- Boissonnat, J.-D., Guibas, L. J., Oudot, S. Y., 2009b. Manifold reconstruction in arbitrary dimensions using witness complexes. *Discrete & Computational Geometry* 42 (1), 37–70.
- Bressan, S., Li, J., Ren, S., Wu, J., 2016. The embedded homology of hypergraphs and applications. *arXiv preprint arXiv:1610.00890*.
- Brüel-Gabrielsson, R., Nelson, B. J., Dwarknath, A., Skraba, P., Guibas, L. J., Carlsson, G., 2019. A topology layer for machine learning. *arXiv preprint arXiv:1905.12200*.
- Bubenik, P., 2015. Statistical topological data analysis using persistence landscapes. *Journal of Machine Learning Research* 16, 77–102.
- Bubenik, P., 2020. The persistence landscape and some of its properties. In: *Topological Data Analysis*. Springer, pp. 97–117.
- Bubenik, P., Carlsson, G., Kim, P., Luo, Z.-M., 2010. Statistical topology via Morse theory persistence and nonparametric estimation. *Algebraic methods in statistics and probability II* 516, 75–92.
- Bubenik, P., Hull, M., Patel, D., Whittle, B., 2020. Persistent homology detects curvature. *Inverse Problems* 36 (2), 025008.
- Bullmore, E., Sporns, O., 2009. Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature Review Neuroscience* 10, 186–98.
- Carlsson, G., Gabrielsson, R. B., 2018. Topological approaches to deep learning. *arXiv preprint arXiv:1811.01122*.
- Carlsson, G., Ishkhanov, T., De Silva, V., Zomorodian, A., 2008. On the local behavior of spaces of natural images. *International journal of computer vision* 76, 1–12.
- Carlsson, G., Memoli, F., 2008. Persistent clustering and a theorem of J. Kleinberg. *arXiv preprint arXiv:0808.2241*.
- Carlsson, G., Memoli, F., 2010. Characterization, stability and convergence of hierarchical clustering methods. *Journal of Machine Learning Research* 11, 1425–1470.
- Cassidy, B., Rae, C., Solo, V., 2015. Brain activity: conditional dissimilarity and persistent homology. In: *IEEE 12th International Symposium on Biomedical Imaging (ISBI)*. pp. 1356–1359.
- Chakraborty, A., Alam, M., Dey, V., Chattopadhyay, A., Mukhopadhyay, D., 2018. Adversarial attacks and defences: A survey. *arXiv preprint arXiv:1810.00069*.
- Chan, G., Wood, A., 1997. Algorithm as 312: An algorithm for simulating stationary Gaussian random fields. *Journal of the Royal Statistical Society: Series C* 46, 171–181.
- Chan, J. M., Carlsson, G., Rabadan, R., 2013. Topology of viral evolution. *Proceedings of the National Academy of Sciences* 110 (46), 18566–18571.
- Chaudhuri, K., Kakade, S., Livescu, K., Sridharan, K., 2009. Multi-view clustering via canonical correlation analysis. In: *Proceedings of the 26th annual international conference on machine learning*. pp. 129–136.
- Chazal, F., Cohen-Steiner, D., Guibas, L., Memoli, F., Oudot, S., 2009. Gromov-Hausdorff stable signatures for shapes using persistence. In: *Computer Graphics Forum*. Vol. 28. pp. 1393–1403.
- Chazal, F., Cohen-Steiner, D., Mérigot, Q., 2011. Geometric inference for probability measures. *Foundations of Computational Mathematics* 11 (6), 733–751.
- Chazal, F., Michel, B., 2017. An introduction to topological data analysis: fundamental and practical aspects for data scientists. *arXiv preprint arXiv:1710.04019*.
- Chen, C., Ni, X., Bai, Q., Wang, Y., 2019. A topological regularizer for classifiers via persistent homology. In: *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 2573–2582.
- Chen, W., Cockrell, C., Ward, K. R., Najarian, K., 2010. Intracranial pressure level prediction in traumatic brain injury by extracting features from multiple sources and using machine learning methods. In: *2010 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, pp. 510–515.
- Chen, X., Zhang, H., Gao, Y., Wee, C.-Y., Li, G., Shen, D., 2016. High-order resting-state functional connectivity network for MCI classification. *Human Brain Mapping* 37, 3282–3296.
- Chung, F., Graham, R., 1992. Cohomological aspects of hypergraphs. *Transactions of the American Mathematical Society* 334 (1), 365–388.
- Chung, M., 2012. *Computational Neuroanatomy: The Methods*. World Scientific, Singapore.
- Chung, M., 2018. Statistical challenges of big brain network data. *Statistics and Probability Letter* 136, 78–82.
- Chung, M., Adluru, N., Dalton, K., Alexander, A., Davidson, R., 2011. Scalable brain network construction on white matter fibers. In: *Proc. of SPIE*. Vol. 7962. p. 79624G.
- Chung, M., Bubenik, P., Kim, P., 2009a. Persistence diagrams of cortical surface data. *Proceedings of the 21st International Conference on Information Processing in Medical Imaging (IPMI)*, Lecture Notes in Computer Science (LNCS) 5636, 386–397.
- Chung, M., Dalton, K., Davidson, R., 2008. Tensor-based cortical surface morphometry via weighted spherical harmonic representation. *IEEE Transactions on Medical Imaging* 27, 1143–1151.
- Chung, M., Hanson, J., Lee, H., Adluru, N., Alexander, A. L., Davidson, R., Pollak, S., 2013. Persistent homological sparse network approach to detect-

- ing white matter abnormality in maltreated children: MRI and DTI multi-modal study. MICCAI, Lecture Notes in Computer Science (LNCS) 8149, 300–307.
- Chung, M., Hanson, J., Ye, J., Davidson, R., Pollak, S., 2015a. Persistent homology in sparse regression and its application to brain morphometry. *IEEE Transactions on Medical Imaging* 34, 1928–1939.
- Chung, M., Huang, S.-G., Gritsenko, A., Shen, L., Lee, H., 2019a. Statistical inference on the number of cycles in brain networks. In: 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019). IEEE, pp. 113–116.
- Chung, M., Lee, H., DiChristofano, A., Ombao, H., Solo, V., 2019b. Exact topological inference of the resting-state brain networks in twins. *Network Neuroscience* 3, 674–694.
- Chung, M., Lee, H., Solo, V., Davidson, R., Pollak, S., 2017a. Topological distances between brain networks. *International Workshop on Connectomics in Neuroimaging*, 161–170.
- Chung, M., Qiu, A., Seo, S., Vorperian, H., 2015b. Unified heat kernel regression for diffusion, kernel smoothing and wavelets on manifolds and its application to mandible growth modeling in CT images. *Medical Image Analysis* 22, 63–76.
- Chung, M., Singh, V., Kim, P., Dalton, K., Davidson, R., 2009b. Topological characterization of signal in brain images using min-max diagrams. MICCAI, Lecture Notes in Computer Science (LNCS) 5762, 158–166.
- Chung, M., Vilalta-Gil, V., Lee, H., Rathouz, P., Lahey, B., Zald, D., 2017b. Exact topological inference for paired brain networks via persistent homology. *Information Processing in Medical Imaging (IPMI)*, Lecture Notes in Computer Science (LNCS) 10265, 299–310.
- Chung, M., Wang, Y., Wu, G., 2018. Heat kernel smoothing in irregular image domains. *International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 5101–5104.
- Chung, M., Xie, L., Huang, S.-G., Wang, Y., Yan, J., Shen, L., 2019c. Rapid acceleration of the permutation test via transpositions. In: *International Workshop on Connectomics in Neuroimaging*. Vol. 11848. Springer, pp. 42–53.
- Clough, J., Oksuz, I., Byrne, N., Schnabel, J., King, A., 2019. Explicit topological priors for deep-learning based image segmentation using persistent homology. In: *International Conference on Information Processing in Medical Imaging*. Springer, pp. 16–28.
- Cohen-Steiner, D., Edelsbrunner, H., Harer, J., 2007. Stability of persistence diagrams. *Discrete and Computational Geometry* 37, 103–120.
- Cole, A., 2020. Identifying and exploiting structure in cosmological and string theoretic data. Ph.D. Thesis, University of Wisconsin-Madison.
- Cole, A., Loges, G. J., Shiu, G., 2020. Quantitative and Interpretable Order Parameters for Phase Transitions from Persistent Homology.
- Cole, A., Shiu, G., 2019. Topological data analysis for the string landscape. *Journal of High Energy Physics* 2019 (3), 54.
- Cole, J. H., Poudel, R. P., Tsagkrasoulis, D., Caan, M. W., Steves, C., Spector, T. D., Montana, G., 2017. Predicting brain age with deep learning from raw imaging data results in a reliable and heritable biomarker. *NeuroImage* 163, 115–124.
- Cootes, T., Hill, A., Taylor, C., Haslam, J., 1993. The use of active shape models for locating structures in medical images. *Lecture Notes in Computer Science* 687, 33–33.
- Correa, N., Li, Y.-O., Adali, T., Calhoun, V., 2009. Fusion of fMRI, sMRI, and EEG data using canonical correlation analysis. In: 2009 IEEE International Conference on Acoustics, Speech and Signal Processing. pp. 385–388.
- Dabaghian, Y., Mémoli, F., Frank, L., Carlsson, G., 2012. A topological paradigm for hippocampal spatial map formation using persistent homology. *PLoS Comput Biol* 8 (8), e1002581.
- Davison, E. N., Schlesinger, K. J., Bassett, D. S., Lynall, M.-E., Miller, M. B., Grafton, S. T., Carlson, J. M., 2015. Brain network adaptability across task states. *PLoS comput biol* 11 (1), e1004029.
- Davison, E. N., Turner, B. O., Schlesinger, K. J., Miller, M. B., Grafton, S. T., Bassett, D. S., Carlson, J. M., 2016. Individual differences in dynamic functional brain connectivity across the human lifespan. *PLoS computational biology* 12 (11), e1005178.
- de Silva, V., Carlsson, G., 2004. Topological estimation using witness complexes. *SPBG* 4, 157–166.
- de Silva, V., Ghrist, R., 2007. Homological sensor networks. *Notic Amer Math Soc* 54, 10–17.
- Desai, R., Liebenthal, E., Possing, E., Waldron, E., Binder, J., 2005. Volumetric vs. surface-based alignment for localization of auditory cortex activation. *NeuroImage* 26 (4), 1019–1029.
- Divol, V., Polonik, W., 2019. On the choice of weight functions for linear representations of persistence diagrams. *Journal of Applied and Computational Topology* 3 (3), 249–283.
- Dong, P., Guo, Y., Shen, D., Wu, G., 2015. Multi-atlas and multi-modal hippocampus segmentation for infant mr brain images by propagating anatomical labels on hypergraph. In: *International Workshop on Patch-based Techniques in Medical Imaging*. Springer, pp. 188–196.
- Edelsbrunner, H., Harer, J., 2008. Persistent homology - a survey. *Contemporary Mathematics* 453, 257–282.
- Edelsbrunner, H., Harer, J., 2010a. *Computational Topology*. American Mathematical Society.
- Edelsbrunner, H., Harer, J., 2010b. *Computational topology: An introduction*. American Mathematical Society.
- Edelsbrunner, H., Letscher, D., Zomorodian, A., 2002. Topological persistence and simplification. *Discrete and Computational Geometry* 28, 511–533.
- Edelsbrunner, H., Mücke, E. P., 1994. Three-dimensional alpha shapes. *ACM Transactions on Graphics (TOG)* 13 (1), 43–72.
- Edmonds, J., Karp, R. M., 1972. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the ACM (JACM)* 19 (2), 248–264.
- Efron, B., 1982. The jackknife, the bootstrap and other resampling plans. Vol. 38. SIAM.
- Eickhoff, S., Nichols, T., Laird, A., Hoffstaedter, F., Amunts, K., Fox, P., Bzdok, D., Eickhoff, C., 2016. Behavior, sensitivity, and power of activation likelihood estimation characterized by massive empirical simulation. *NeuroImage* 137, 70–85.
- Emtander, E., 2009. Betti numbers of hypergraphs. *Communications in algebra* 37 (5), 1545–1571.
- Estrada, E., Rodriguez-Velazquez, J. A., 2005. Complex networks as hypergraphs. *arXiv preprint physics/0505137*.
- Fisher, R., 1966. *The Design of Experiments*, 8th Edition. Hafner.
- Fong, R. C., Scheirer, W. J., Cox, D. D., 2018. Using human brain activity to guide machine learning. *Scientific reports* 8 (1), 1–10.
- Forman, R., 1998. Morse theory for cell complexes.
- Friedman, J., 1998. Computing betti numbers via combinatorial laplacians. *Algorithmica* 21 (4), 331–346.
- Gabella, M., 2019. Topology of learning in artificial neural networks. *arXiv e-prints*, arXiv:1902.
- Gabrielsson, R. B., Carlsson, G., 2019. Exposition and interpretation of the topology of neural networks. In: 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA). IEEE, pp. 1069–1076.
- Gameiro, M., Hiraoka, Y., Izumi, S., Kramar, M., Mischaikow, K., Nanda, V., 2015. A topological measurement of protein compressibility. *Japan Journal of Industrial and Applied Mathematics* 32, 1–17.
- Gao, X. W., Hui, R., Tian, Z., 2017. Classification of ct brain images based on deep learning networks. *Computer methods and programs in biomedicine* 138, 49–56.
- Garg, A., Lu, D., Popuri, K., Beg, M., 2017. Brain geometry persistent homology marker for parkinson's disease. In: IEEE 14th International Symposium on Biomedical Imaging. IEEE, pp. 525–528.
- Garin, A., Tauzin, G., 2019. A topological "reading" lesson: classification of MNIST using TDA. In: 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA). IEEE, pp. 1551–1556.
- Garside, K., Gjoka, A., Henderson, R., Johnson, H., Makarenko, I., 2020. Event history and topological data analysis. *arXiv preprint arXiv:2012.08810*.
- Ghrist, R., 2008. Barcodes: The persistent topology of data. *Bulletin of the American Mathematical Society* 45, 61–75.
- Gibbons, J. D., Chakraborti, S., 2011. *Nonparametric Statistical Inference*. Chapman & Hall/CRC Press.
- Gu, S., Yang, M., Medaglia, J. D., Gur, R. C., Gur, R. E., Satterthwaite, T. D., Bassett, D. S., 2017. Functional hypergraph uncovers novel covariant structures over neurodevelopment. *Human brain mapping* 38 (8), 3823–3835.
- Guibas, L., Oudot, S., 2008. Reconstruction using witness complexes. *Discrete & Computational Geometry* 40, 325–356.
- Guo, W., Manohar, K., Brunton, S., Banerjee, A., 2018. Sparse-TDA: Sparse realization of topological data analysis for multi-way classification. *IEEE Transactions on Knowledge and Data Engineering* 30, 1403–1408.
- Guo, X., Srivastava, A., 2020. Representations, metrics and statistics for shape analysis of elastic graphs. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. pp. 832–833.

- Harker, S., Mischaikow, K., Mrozek, M., Nanda, V., Wagner, H., Juda, M., Dlotko, P., 2010. The efficiency of a homology algorithm based on discrete morse theory and coreductions. *Image-A: Applicable Mathematics in Image Engineering*, 1 (1), 41–48.
- Hart, J., 1999. Computational topology for shape modeling. In: *Proceedings of the International Conference on Shape Modeling and Applications*. pp. 36–43.
- He, Y., Chen, Z., Evans, A., 2008. Structural insights into aberrant topological patterns of large-scale cortical networks in Alzheimer’s disease. *Journal of Neuroscience* 28, 4756.
- Heinsfeld, A. S., Franco, A. R., Craddock, R. C., Buchweitz, A., Meneguzzi, F., 2018. Identification of autism spectrum disorder using deep learning and the abide dataset. *NeuroImage: Clinical* 17, 16–23.
- Heo, G., Gamble, J., Kim, P., 2012. Topological analysis of variance and the maxillary complex. *Journal of the American Statistical Association* 107, 477–492.
- Hofer, C., Graf, F., Rieck, B., Niethammer, M., Kwitt, R., 2020. Graph filtration learning. In: *International Conference on Machine Learning*. PMLR, pp. 4314–4323.
- Hofer, C. D., Kwitt, R., Niethammer, M., 2019. Learning representations of persistence barcodes. *Journal of Machine Learning Research* 20 (126), 1–45.
- Horak, D., Maletić, S., Rajković, M., 2009. Persistent homology of complex networks. *Journal of Statistical Mechanics: Theory and Experiment* 2009, P03034.
- Hotelling, H., 1992. Relations between two sets of variates. In: *Breakthroughs in statistics*. Springer, pp. 162–190.
- Hu, X., Li, F., Samaras, D., Chen, C., 2019. Topology-preserving deep image segmentation. In: *Advances in Neural Information Processing Systems*. pp. 5657–5668.
- Huang, S., Papernot, N., Goodfellow, I., Duan, Y., Abbeel, P., 2017. Adversarial attacks on neural network policies. *arXiv:1702.02284*.
- Huang, S.-G., Chung, M. K., Carroll, I. C., Goldsmith, H. H., 2019a. Dynamic functional connectivity using heat kernel. In: *2019 IEEE Data Science Workshop (DSW)*. pp. 222–226.
- Huang, S.-G., Lyu, I., Qiu, A., Chung, M., 2019b. Fast polynomial approximation to heat diffusion in manifolds. *MICCAI* 11767, 48–56.
- Huang, S.-G., Lyu, I., Qiu, A., Chung, M., 2020a. Fast polynomial approximation of heat kernel convolution on manifolds and its application to brain sulcal and gyral graph pattern analysis. *IEEE Transactions on Medical Imaging* 39, 2201–2212.
- Huang, S.-G., Samdin, S.-T., Ting, C., Ombao, H., Chung, M., 2020b. Statistical model for dynamically-changing correlation matrices with application to brain connectivity. *Journal of Neuroscience Methods* 331, 108480.
- Jain, A. K., Murty, M. N., Flynn, P. J., 1999. Data clustering: a review. *ACM computing surveys (CSUR)* 31 (3), 264–323.
- Jie, B., Wee, C.-Y., Shen, D., Zhang, D., 2016. Hyper-connectivity of functional networks for brain disease diagnosis. *Medical image analysis* 32, 84–100.
- Jnawali, K., Arbabshirani, M. R., Rao, N., Patel, A. A., 2018. Deep 3d convolution neural network for ct brain hemorrhage classification. In: *Medical Imaging 2018: Computer-Aided Diagnosis*. Vol. 10575. International Society for Optics and Photonics, p. 105751C.
- Kališnik, S., 2019. Tropical coordinates on the space of persistence barcodes. *Foundations of Computational Mathematics* 19 (1), 101–129.
- Kerber, M., Morozov, D., Nigmatov, A., 2017. Geometry helps to compare persistence diagrams. *Journal of Experimental Algorithmics (JEA)* 22, 1–20.
- Khalid, A., Kim, B., Chung, M., Ye, J., Jeon, D., 2014. Tracing the evolution of multi-scale functional networks in a mouse model of depression using persistent brain network homology. *NeuroImage* 101, 351–363.
- Kiebel, S., Poline, J.-P., Friston, K., Holmes, A., Worsley, K., 1999. Robust smoothness estimation in statistical parametric maps using standardized residuals from the general linear model. *NeuroImage* 10, 756–766.
- Kim, K. H., Choi, S. H., Park, S.-H., 2018. Improving arterial spin labeling by using deep learning. *Radiology* 287 (2), 658–666.
- Kim, W., Adluru, N., Chung, M., Okonkwo, O., Johnson, S., Bendlin, B., Singh, V., 2015. Multi-resolution statistical analysis of brain connectivity graphs in preclinical Alzheimer’s disease. *NeuroImage* 118, 103–117.
- Kondor, R., Howard, A., Jebara, T., 2007. Multi-object tracking with representations of the symmetric group. In: *International Conference on Artificial Intelligence and Statistics (AISTATS)*. Vol. 1. p. 5.
- Kong, R., Gao, J., Xu, Y., Pan, Y., Wang, J., Liu, J., 2019. Classification of autism spectrum disorder by combining brain connectivity and deep neural network classifier. *Neurocomputing* 324, 63–68.
- Kottaimalai, R., Rajasekaran, M. P., Selvam, V., Kannapiran, B., 2013. Eeg signal classification using principal component analysis with neural network in brain computer interface applications. In: *2013 IEEE International Conference ON Emerging Trends in Computing, Communication and Nanotechnology (ICECCN)*. IEEE, pp. 227–231.
- Lee, H., Chung, M.K., K. H., Lee, D., 2014. Hole detection in metabolic connectivity of Alzheimer’s disease using k-Laplacian. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, Lecture Notes in Computer Science. pp. 297–304.
- Lee, H., Chung, M., Choi, H., K. H., Ha, S., Kim, Y., Lee, D., 2019. Harmonic holes as the submodules of brain network and network dissimilarity. *International Workshop on Computational Topology in Image Context, Lecture Notes in Computer Science*, 110–122.
- Lee, H., Chung, M., Kang, H., Choi, H., Kim, Y., Lee, D., 2018. Abnormal hole detection in brain connectivity by kernel density of persistence diagram and Hodge Laplacian. In: *IEEE International Symposium on Biomedical Imaging (ISBI)*. pp. 20–23.
- Lee, H., Chung, M., Kang, H., Kim, B.-N., Lee, D., 2011a. Computing the shape of brain networks using graph filtration and Gromov-Hausdorff metric. *MICCAI, Lecture Notes in Computer Science* 6892, 302–309.
- Lee, H., Chung, M., Kang, H., Kim, B.-N., Lee, D., 2011b. Discriminative persistent homology of brain networks. In: *IEEE International Symposium on Biomedical Imaging (ISBI)*. pp. 841–844.
- Lee, H., Kang, H., Chung, M., Kim, B.-N., Lee, D., 2012. Persistent brain network homology from the perspective of dendrogram. *IEEE Transactions on Medical Imaging* 31, 2267–2277.
- Lee, H., Kang, H., Chung, M., Lim, S., Kim, B.-N., Lee, D., 2017. Integrated multimodal network approach to PET and MRI based on multidimensional persistent homology. *Human Brain Mapping* 38, 1387–1402.
- Lee, H., Lee, D., Kang, H., Kim, B.-N., Chung, M., 2011c. Sparse brain network recovery under compressed sensing. *IEEE Transactions on Medical Imaging* 30, 1154–1165.
- Li, Y., Wang, D., Ascoli, G., Mitra, P., Wang, Y., 2017. Metrics for comparing neuronal tree shapes based on persistent homology. *PloS one* 12 (8), e0182184.
- Lloyd, S., Garnerone, S., Zanardi, P., 2016. Quantum algorithms for topological and geometric analysis of data. *Nature communications* 7 (1), 1–7.
- Mazumder, R., Hastie, T., 2012. Exact covariance thresholding into connected components for large-scale graphical LASSO. *The Journal of Machine Learning Research* 13, 781–794.
- Merelli, E., Piangerelli, M., Rucco, M., Toller, D., 2016. A topological approach for multivariate time series characterization: the epileptic brain. In: *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS)*. pp. 201–204.
- Merelli, E., Rucco, M., Sloot, P., Tesei, L., 2015. Topological characterization of complex systems: Using persistent entropy. *Entropy* 17 (10), 6872–6892.
- Milnor, J., 1973. *Morse Theory*. Princeton University Press.
- Mischaikow, K., Nanda, V., 2013. Morse theory for filtrations and efficient computation of persistent homology. *Discrete & Computational Geometry* 50 (2), 330–353.
- Mittal, M., Goyal, L. M., Kaur, S., Kaur, I., Verma, A., Hemanth, D. J., 2019. Deep learning based enhanced tumor segmentation approach for mr brain images. *Applied Soft Computing* 78, 346–354.
- Moeskops, P., Veta, M., Lafarge, M. W., Eppenhof, K. A., Pluim, J. P., 2017. Adversarial training and dilated convolutions for brain mri segmentation. In: *Deep learning in medical image analysis and multimodal learning for clinical decision support*. Springer, pp. 56–64.
- Muhammad, A., Egerstedt, M., 2006. Control using higher order laplacians in network topologies. In: *Proc. of 17th International Symposium on Mathematical Theory of Networks and Systems*. pp. 1024–1038.
- Osher, S., Fedkiw, R., 2003. *Level set methods and dynamic implicit surfaces*. Springer Verlag.
- O’Sullivan, F., Saha, A., 1999. Use of ridge regression for improved estimation of kinetic constants from pet data. *IEEE transactions on medical imaging* 18 (2), 115–125.
- Otter, N., Porter, M., Tillmann, U., Grindrod, P., Harrington, H., 2017. A roadmap for the computation of persistent homology. *EPJ Data Science*

- Pachauri, D., Hinrichs, C., Chung, M., Johnson, S., Singh, V., 2011. Topology based kernels with application to inference problems in alzheimer's disease. *IEEE Transactions on Medical Imaging*, 1760–1770.
- Palande, S., Jose, V., Zielinski, B., Anderson, J., Fletcher, P., Wang, B., 2017. Revisiting abnormalities in brain network architecture underlying autism using topology-inspired statistical inference. In: *International Workshop on Connectomics in Neuroimaging*. pp. 98–107.
- Petri, G., Expert, P., Turkheimer, F., Carhart-Harris, R., Nutt, D., Hellyer, P., Vaccarino, F., 2014. Homological scaffolds of brain functional networks. *Journal of The Royal Society Interface* 11, 20140873.
- Petri, G., Scolamiero, M., Donato, I., Vaccarino, F., 2013. Topological strata of weighted complex networks. *PLoS One* 8, e66506.
- Pothen, A., Fan, C., 1990. Computing the block triangular form of a sparse matrix. *ACM Transactions on Mathematical Software (TOMS)* 16, 324.
- Pun, C. S., Xia, K., Lee, S. X., 2018. Persistent-homology-based machine learning and its applications—a survey. *arXiv preprint arXiv:1811.00252*.
- Qiu, A., Lee, A., Tan, M., Chung, M., 2015. Manifold learning on brain functional networks in aging. *Medical image analysis* 20, 52–60.
- Rathore, A., Palande, S., Anderson, J. S., Zielinski, B. A., Fletcher, P. T., Wang, B., 2019. Autism classification using topological features and deep learning: A cautionary tale. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, pp. 736–744.
- Reininghaus, J., Huber, S., Bauer, U., Kwitt, R., 2015. A stable multi-scale kernel for topological machine learning. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 4741–4748.
- Rottschy, C., Langner, R., Dogan, I., Reetz, K., Laird, A., Schulz, J., Fox, P., Eickhoff, S., 2012. Modelling neural correlates of working memory: a coordinate-based meta-analysis. *NeuroImage* 60, 830–846.
- Rubinov, M., Knock, S. A., Stam, C. J., Micheloyannis, S., Harris, A. W., Williams, L. M., Breakspear, M., 2009. Small-world properties of nonlinear brain activity in schizophrenia. *Human Brain Mapping* 30, 403–416.
- Rubinov, M., Sporns, O., 2010. Complex network measures of brain connectivity: Uses and interpretations. *NeuroImage* 52, 1059–1069.
- Rucco, M., Castiglione, F., Merelli, E., Pettini, M., 2016. Characterisation of the idiosyncratic immune network through persistent entropy. In: *Proceedings of ECCS 2014*. Springer, pp. 117–128.
- Salvador, R., Suckling, J., Coleman, M. R., Pickard, J. D., Menon, D., Bullmore, E., 2005. Neurophysiological architecture of functional magnetic resonance images of human brain. *Cerebral Cortex* 15, 1332–1342.
- Sato, Y., Nakajima, S., Shiraga, N., Atsumi, H., Yoshida, S., Koller, T., Gerig, G., Kikinis, R., 1998. Three-dimensional multi-scale line filter for segmentation and visualization of curvilinear structures in medical images. *Medical Image Analysis* 2, 143–168.
- Schaub, M., Benson, A., Horn, P., Lippner, G., Jadbabaie, A., 2018. Random walks on simplicial complexes and the normalized hodge laplacian. *arXiv preprint arXiv:1807.05044*.
- Sheehy, D. R., 2013. Linear-size approximations to the vietoris–rips filtration. *Discrete & Computational Geometry* 49 (4), 778–796.
- Si, H., 2015. TetGen, a Delaunay-based quality tetrahedral mesh generator. *ACM Transactions on Mathematical Software (TOMS)* 41, 1–36.
- Singh, G., Memoli, F., Ishkhanov, T., Sapiro, G., Carlsson, G., Ringach, D., 2008. Topological analysis of population activity in visual cortex. *Journal of Vision* 8, 1–18.
- Solo, V., Poline, J., Lindquist, M., Simpson, S., Bowman, D., C. M., Cassidy, B., 2018. Connectivity in fMRI: a review and preview. *IEEE Transactions on Medical Imaging*, in press.
- Songdechakraiwut, T., Chung, M., 2020a. Dynamic topological data analysis for functional brain signals. *IEEE International Symposium on Biomedical Imaging Workshops*, 1–4.
- Songdechakraiwut, T., Chung, M., 2020b. Topological learning for brain networks, *arXiv:2012.00675*.
- Stolz, B., Emerson, T., Nahkuri, S., Porter, M., Harrington, H., 2018. Topological data analysis of task-based fMRI data from experiments on schizophrenia. *arXiv preprint arXiv:1809.08504*.
- Stolz, B., Harrington, H., Porter, M., 2017. Persistent homology of time-dependent functional networks constructed from coupled time series. *Chaos: An Interdisciplinary Journal of Nonlinear Science* 27, 047410.
- Suk, H.-I., Lee, S.-W., Shen, D., Initiative, A. D. N., et al., 2017. Deep ensemble learning of sparse regression models for brain disease diagnosis. *Medical image analysis* 37, 101–113.
- Supekar, K., Menon, V., Rubin, D., Musen, M., Greicius, M., 2008. Network analysis of intrinsic functional brain connectivity in Alzheimer's disease. *PLoS Computational Biology* 4 (6), e1000100.
- Taylor, J., Worsley, K., 2008. Random fields of multivariate test statistics, with applications to shape analysis. *Annals of Statistics* 36, 1–27.
- Thirion, B., Varoquaux, G., Dohmatob, E., Poline, J.-B., 2014. Which fMRI clustering gives good brain parcellations? *Frontiers in neuroscience* 8, 167.
- Topaz, C., Ziegelmeier, L., Halverson, T., 2015. Topological data analysis of biological aggregation models. *PLoS One*, e0126383.
- Torres, L., Blevins, A. S., Bassett, D. S., Eliassi-Rad, T., 2020. The why, how, and when of representations for complex systems. *arXiv preprint arXiv:2006.02870*.
- Turner, K., Mileyko, Y., Mukherjee, S., Harer, J., 2014. Fréchet means for distributions of persistence diagrams. *Discrete & Computational Geometry* 52 (1), 44–70.
- Tuzhilin, A., 2016. Who invented the Gromov-Hausdorff distance? *arXiv preprint arXiv:1612.00728*.
- Uddin, L., Kelly, A., Biswal, B., Margulies, D., Shehzad, Z., Shaw, D., Ghafari, M., Rotrosen, J., Adler, L., Castellanos, F., Milham, M., 2008. Network homogeneity reveals decreased integrity of default-mode network in ADHD. *Journal of Neuroscience Methods* 169, 249–254.
- Wang, G., Wang, Y., Initiative, A. D. N., 2017a. Towards a holistic cortical thickness descriptor: heat kernel-based grey matter morphology signatures. *Neuroimage* 147, 360–380.
- Wang, H., Nie, F., Huang, H., Risacher, S., Ding, C., Saykin, A. J., Shen, L., 2011. Sparse multi-task regression and feature selection to identify brain imaging predictors for memory performance. In: *2011 International Conference on Computer Vision. IEEE*, pp. 557–562.
- Wang, Y., Chung, M., Denticio, D., Lutz, A., Davidson, R., 2017b. Topological network analysis of electroencephalographic power maps. In: *International Workshop on Connectomics in NeuroImaging, Lecture Notes in Computer Science (LNCS)*. Vol. 10511. pp. 134–142.
- Wang, Y., Ombao, H., Chung, M., 2018. Topological data analysis of single-trial electroencephalographic signals. *Annals of Applied Statistics* 12, 1506–1534.
- Wang, Y., Ombao, H., Chung, M., 2019. Statistical persistent homology of brain signals. *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1125–1129.
- Wijk, B. C. M., Stam, C. J., Daffertshofer, A., 2010. Comparing brain networks of different size and connectivity density using graph theory. *PloS one* 5, e13701.
- Wolterink, J. M., Dinkla, A. M., Savenije, M. H., Seevinck, P. R., van den Berg, C. A., Išgum, I., 2017. Deep mr to ct synthesis using unpaired data. In: *International workshop on simulation and synthesis in medical imaging*. Springer, pp. 14–23.
- Wong, E., Palande, S., Wang, B., Zielinski, B., Anderson, J., Fletcher, P., 2016. Kernel partial least squares regression for relating functional brain network topology to clinical measures of behavior. In: *IEEE International Symposium on Biomedical Imaging (ISBI)*. pp. 1303–1306.
- Worsley, K., Marrett, S., Neelin, P., Vandal, A., Friston, K., Evans, A., 1996. A unified statistical approach for determining significant signals in images of cerebral activation. *Human Brain Mapping* 4, 58–73.
- Xu, X., Cisewski-Kehe, J., Green, S. B., Nagai, D., 2019. Finding cosmic voids and filament loops using topological data analysis. *Astron. Comput.* 27, 34–52.
- Yoo, K., Lee, P., Chung, M., Sohn, W., Chung, S., Na, D., Ju, D., Jeong, Y., 2017. Degree-based statistic and center persistency for brain connectivity analysis. *Human Brain Mapping* 38, 165–181.
- Zavlanos, M., Pappas, G., 2008. A dynamical systems approach to weighted graph matching. *Automatica* 44, 2817–2824.
- Zhang, G., Cai, B., Zhang, A., Stephen, J., Wilson, T., Calhoun, V., Wang, Y.-P., 2019. Estimating Dynamic Functional Brain Connectivity With a Sparse Hidden Markov Model. *IEEE transactions on medical imaging* 39, 488–498.
- Zhang, Y. C., Kagen, A. C., 2017. Machine learning interface for medical image analysis. *Journal of digital imaging* 30 (5), 615–621.
- Zhu, X., Suk, H.-I., Shen, D., 2014. Matrix-similarity based loss function and feature selection for Alzheimer's disease diagnosis. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. pp. 3089–3096.
- Zomorodian, A., 2009. *Topology for computing*. Vol. 16 of Cambridge Monographs on Applied and Computational Mathematics. Cambridge University



Press, Cambridge.

Zomorodian, A., Carlsson, G., 2005. Computing persistent homology. *Discrete and Computational Geometry* 33, 249–274.

Zu, C., Gao, Y., Munsell, B., Kim, M., Peng, Z., Zhu, Y., Gao, W., Zhang, D., Shen, D., Wu, G., 2016. Identifying high order brain connectome biomarkers via learning on hypergraph. In: *International Workshop on Machine Learning in Medical Imaging*. Springer, pp. 1–9.