UNIVERSITÉ CÔTE D'AZUR

MSc. DSAI

Topological Data Analysis of Human Brain Data

Internship report

June 2022

par

Ufuk Cem Birbiri

Academic tutor: Mathieu Desroches

Inria Center at UCA

Inria

Table des matières

Abstract

Topology is a subfield of mathematics that analyses the shapes of objects. Topological data analysis is a new method to discover underlying patterns in a given dataset using topological methods. The Mapper algorithm is one of the topological analysis technique which is used in mapping data points to a predetermined filter space. Through this method, we often find subgroups in data sets that traditional methodologies fail to find. In this internship, we have analysed a dataset consisting of 998 patients with their neurological variables. There are cognitive, motor, emotional, and personal information (age and gender) of patients in the dataset. Also, the brain connectivity matrices are used as a summary of their brain functions. To get the important features of these matrices, we used the method of Sparse-Rips complex to extract features. The Mapper algorithm then converts the data into a topological complex called a "mapper complex". The topological features of the mapper complex gives us the subgroups of patients. In our study, Mapper created 3 female and one male groups that have different characteristics. Through the topological analysis, we found that patients within the group found have problems of sadness and loneliness. Likewise, we found that patients within the female groups perform better at language, and cognitive tasks than patients within the male group.

1 Introduction

1.1 What is Topological Data Analysis (TDA) ?

Data gathering and analyses become a fundamental power in all research communities such as medicine, engineering, social sciences, economy or mathematics, to name but a few. The amount of data gathered from different areas grows in a gigantic rate therefore it is essential to find a scientific methods to analyse them and extract from them meaningful knowledge. Understanding the *shape of data* brings us to topology, which is a branch of mathematics. Topology is the field of mathematics that studies how space is "connected". It was first studied by Swiss mathematician Leonhard Euler [1] in the 18th century. Over the last 20 years, topological techniques have been used in various applied problems, one of these fields is called Topological Data Analysis (TDA). TDA provides detailed information about the characteristics of the data by providing some topological, statistical and geometrical methods to infer complex topological structures such as connected components, holes, branches or loops. This analysis method is very robust to outliers and noise where the data is usually represented as point clouds or distance matrices in a Euclidean metric space.

1.2 Applications of TDA in Data Science.

There have been many promising results in recent years for applying topological and geometric approaches in different areas. TDA was used in material sciences [2, 3], shape analysis of 3D objects [4, 5], analysis of images in various domains [6, 7], time series analysis [8, 9, 10], medicine [11, 12], biology [11], genomic data [13, 14], or chemistry [15]. Topological approaches and how to use them in interdisciplinary research topics and data science is currently a very active research domain.

In the last few years, there has been a great effort to implement robust TDA structures and algorithms to make them available to the scientific community. These TDA libraries are publicly available and easy to use for any research field. The most popular ones include GUDHI (Python and C++) [16], Scikit-TDA [17], or giotto-tda [18].

One of the methods in topological analysis of complex data is the *Mapper algorithm*, which is very much used in research. The Mapper algorithm uses some metrics and lenses to convert data into a topological network where nodes represent similar data points and edges represent topological relations. It is used for visualization of high-dimensional data and for clustering the data points to extract similarities between communities.

During this internship, I have applied the Mapper algorithm to data that were obtained from the UCLA multi-modal connectivity database [19]. This dataset has many features of the human brain such as cognitive behavior, motor skills, or brain images of patients. My goal was to apply Mapper to this complex dataset to (i) visualize, (ii) find similarities between data points to extract different communities, and (iii) understand what features make these communities unique. The outcome of the Mapper algorithm gives a graph that one can visualize in 2D or 3D. I then applied some statistical tests to find unique communities with their features in the topological network. Overall, the TDA pipeline I have put in place during my internship can be described through the following steps :

1. Preprocessing of the data ;
2. Applying the Mapper algorithm and extracting the low-dimensional shape of data points, which is the visualization result obtained from Mapper ;
3. Detecting communities from the result of Mapper ;
4. Discovering what features make these communities unique.

The remainder of this internship report provides details about the dataset I have worked with, the Mapper algorithm, and the TDA pipeline.

2 Dataset and Preprocessing

The dataset belongs to the NIH Human Connectome Project (HCP) released in 2009 in the Blueprint Grand Challenge [19]. The purpose of the Human Connectome Project is to discover the structure of the human brain and its functional connectivity. It brings many advances in neuroscience research by shedding new light onto key questions such as : how the brain circuitry changes as humans age, how it changes from the psychiatric and neurological standpoints, and how the electrical signals generate thought, feelings, or human behavior. To date, there have been over 100 publications that use HCP data since the first publicly available release [20]. Overall, HCP data helps us to discover what makes humans unique and leads to great advances in the understanding of the human brain in particular in link with neurological and psychiatric disorders.

The dataset includes 998 patients with various features such as brain scans (diffusion tensor images), personal information (family history, personality, age, gender, etc.), cognitive behavior, motor skills, emotion, sensory (vision, taste, pain, audition, olfaction) and more. The correlation matrix of the data columns is shown in Fig. 9 in the Appendix

(section 5). It seems that there is no high correlation between features except strength and gender. Strength and gender are negatively correlated. In my internship work, I have used the following categories in the data :

- **Diffusion Tensor Image (Connectivity matrices)**
- **Personal information :** Age and gender
- **Cognition :** Episodic memory(Picture Sequence Memory), Executive function : Cognitive flexibility(Dimensional Change Card Sort) and inhibition (Flanker Task), Fluid Intelligence(Penn Progressive Matrices), Language/Reading Decoding (Oral Reading Recognition), Language/Vocabulary Comprehension (Picture Vocabulary), Processing Speed (Pattern Completion Processing Speed), Self-regulation/Impulsivity (Delay Discounting), Spatial Orientation (Variable Short Penn Line Orientation Test), Sustained Attention (Short Penn Continuous Performance Test), Verbal Episodic Memory (Penn Word Memory Test), Working Memory (List Sorting)
- **Emotion :** Emotion recognition(Penn Emotion Recognition Test), negative affect, psychological well-being, social relationships, stress and self efficacy
- **Motor :** Endurance(2 minute walk test), locomotion (4-meter walk test), dexterity (9-hole Pegboard), strength (Grip Strength Dynamometry)

2.1 Preprocessing

Most of the features in the dataset have scalar data types. Only connectivity matrices of patients' brain scans have graph-type data and some personal information (gender and age) is of string type. I used the Sklearn categorical encoders [21] to convert string type to scalar.

Diffusion Tensor Image (Connectivity matrices)

In the dataset there are brain scans for each patient in the form of diffusion tensor images (DTI). DTI is a popular brain imaging technique that measures white matter of the brain i.e. the diffusion of water molecules. For the sake of simplicity, I used the converted version of DT images, that is, connectivity matrices. Each image has an equal non-symmetric connectivity matrix. I do not know how DT images were converted into connectivity matrices.

Connectivity matrices measure the amount of fibers from one region to another region of the brain. There is one matrix for each patient and each represents a weighted directed graph. Feature extraction was done separately for each patient. There are 116 brain regions

represented so this is a 116-by-116 matrix. The rows and columns correspond to brain region names. An example of connectivity matrix is shown in Fig. 1 below.

data.1	Precentral_L	Precentral_R	Frontal_Sup_L	Frontal_Sup_R
Precentral_L	0.000000	0.111284	0.244940	0.036833
Precentral_R	0.058161	0.000000	0.024979	0.077539
Frontal_Sup_L	0.638208	0.124529	0.000000	0.278553
Frontal_Sup_R	0.077435	0.311908	0.224755	0.000000
Frontal_Sup_Orb_L	0.000000	0.000000	0.031933	0.002034
...	...	...	...	...

FIGURE 1 – *An example of connectivity matrix of a patient in shape. Rows and columns represent brain regions.*

If we can find a good way to extract features from connectivity matrices, then we can extract underlying knowledge in the diffusion images and add them to the final dataset. There are several ways to extract information from graphs such as : (i) flattening the matrices and using dimension reduction methods such as, e.g., principal component analysis (PCA) or singular value decomposition (SVD), (ii) using graph metrics (density, degree/strength, centrality, etc), (iii) using topological feature extraction methods. Which methods one selects has a strong influence on the outcome, so the choice must be wise. At this point, we should analyze our graph data where all graphs are weighted, directed, and sparse. They are sparse because, in the matrices, most of the entries are 0, which means there is no direct fiber between these two regions. Sparse graphs would also have different shapes. Using TDA can be a good idea because topology is the field of analyzing the shape of objects. We have different shaped graphs in the 116-dimensional matrices, so using topology again seems appropriate.

A topological feature can be a connected component, a 1D hole/loop, a 2D cavity, or more generally a d-dimensional "void". These features are based on the shape of the data. Firstly, graphs are converted to persistence diagrams via a persistence complex. There are five complexes in the giotto-tda [18] library such as *Vietoris-Rips, Weighted-Rips, Sparse-Rips, Weak-Alpha, and Euclidean-Čech persistence.* Since we have weighted, undirected and sparse graphs, Sparse-Rips persistence [22] was the most suitable one. After the complex computation, the graphs turn into persistence diagrams. There is one persistence diagram for each patient in the dataset calculated from the Sparse-Rips complex.

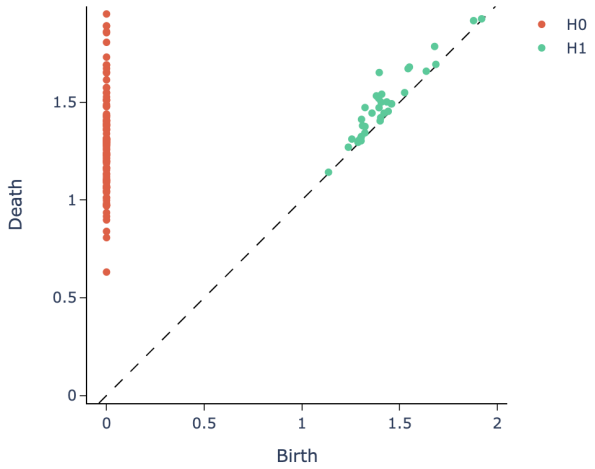

FIGURE 2 – *An example of a persistence diagram as an output of Sparse-Rips complex. The red points are the topological features in dimension 0, and the green points are the topological features in dimension 1. The diagram belongs to one of the patient in the dataset.*

Persistence diagrams are representations of topological features. Each point is a topological feature that appears between a birth and a death scale in the diagram. A point's distance from the diagonal visually represents how "persistent" the associated topological feature is. For example, in Fig. 2 topological features are shown using different colors. In dimension 0, the features are represented in red, and in dimension 1 they are represented in green. In dimension 0, the diagram illustrates the connectivity structure of the graph. In dimension 1, it illustrates the independent one-dimensional holes.
It is also possible to convert the information contained in persistence diagrams into scalar features using *Persistence Entropy* transformer [23] in giotto-tda. From a practical viewpoint, whenever there was a NaN value in a column, I replaced it with the mean value of the corresponding column. In the end, I have obtained a dataset with 29 features, shaped in a (998,29) matrix that is shown in Fig. 3 below.

3 The Mapper Algorithm

Mapper is a computational method for complex datasets to extract the simplest information in the form of simplicial complexes that conserve the underlying topological structures from the original data. It was first implemented in 2007 [24], and since then it has become a popular visualization tool in topological data analysis. It is used for the

	TDA_feature_1	TDA_feature_2	Age	Gender	PicSeq_AgeAdj	CardSort_AgeAdj	Flanker_AgeAdj	PMAT24_A_CR	ReadEng_AgeAdj	PicVocab_AgeAdj	...
0	6.819667	4.526492	1	1	118.78	104.94	116.55	20.0	103.44410	117.03610	...
1	6.821509	4.723720	1	2	103.45	109.92	101.90	17.0	98.73000	96.81000	...
2	6.817826	4.452232	2	1	125.19	100.77	113.51	7.0	125.64000	132.63000	...
3	6.820391	4.500798	1	1	101.69	115.18	114.18	23.0	132.41240	146.59710	...
4	6.824610	4.486151	2	2	70.00	94.30	92.33	11.0	101.16970	69.45302	...
5	6.819778	4.678398	3	1	97.37	105.69	96.19	14.0	112.97570	116.96990	...

FIGURE 3 – *Final dataset with topological feature columns. The end shape of the dataset is (998,29).*

visualization of high-dimensional datasets, simplification, and qualitative analysis. The idea is to partially cluster the data that is guided by scalar filter functions. The resulted visualization is a simple collapse of data into a low-dimensional graph, where the filter function (*lens*) acts as guidance. I used the Mapper algorithm implemented in the GUDHI library [16].

The Mapper algorithm has various steps, which can be seen in Fig. 7 and that are explained in following.

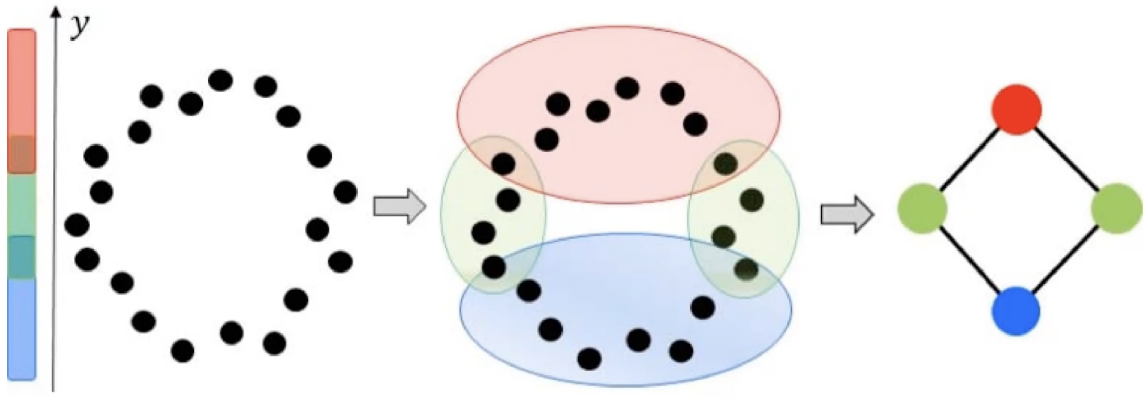

FIGURE 4 – *A visualization of the Mapper algorithm. The chosen filter is the y-axis. The image is taken from [25].*

3.1 Selecting the filter function

The key idea of the Mapper algorithm is to map data points into a metric space using some filters defined on data points while capturing topological and geometric information at a specified resolution and gain. The filter function (also called a lens) is the metric that allows to map our data points. It should be a scalar vector whose size equals the number of data points in the dataset. For example, in the case studied here, the shape of the final dataset is (998,29), so the size of one filter function must be (998,1). Filter functions are calculated based on the dataset. Each entry in a filter vector represents the cover of an individual data point. Each row in a dataset is converted to a single number, i.e., each row provides a real number in the filter vector.

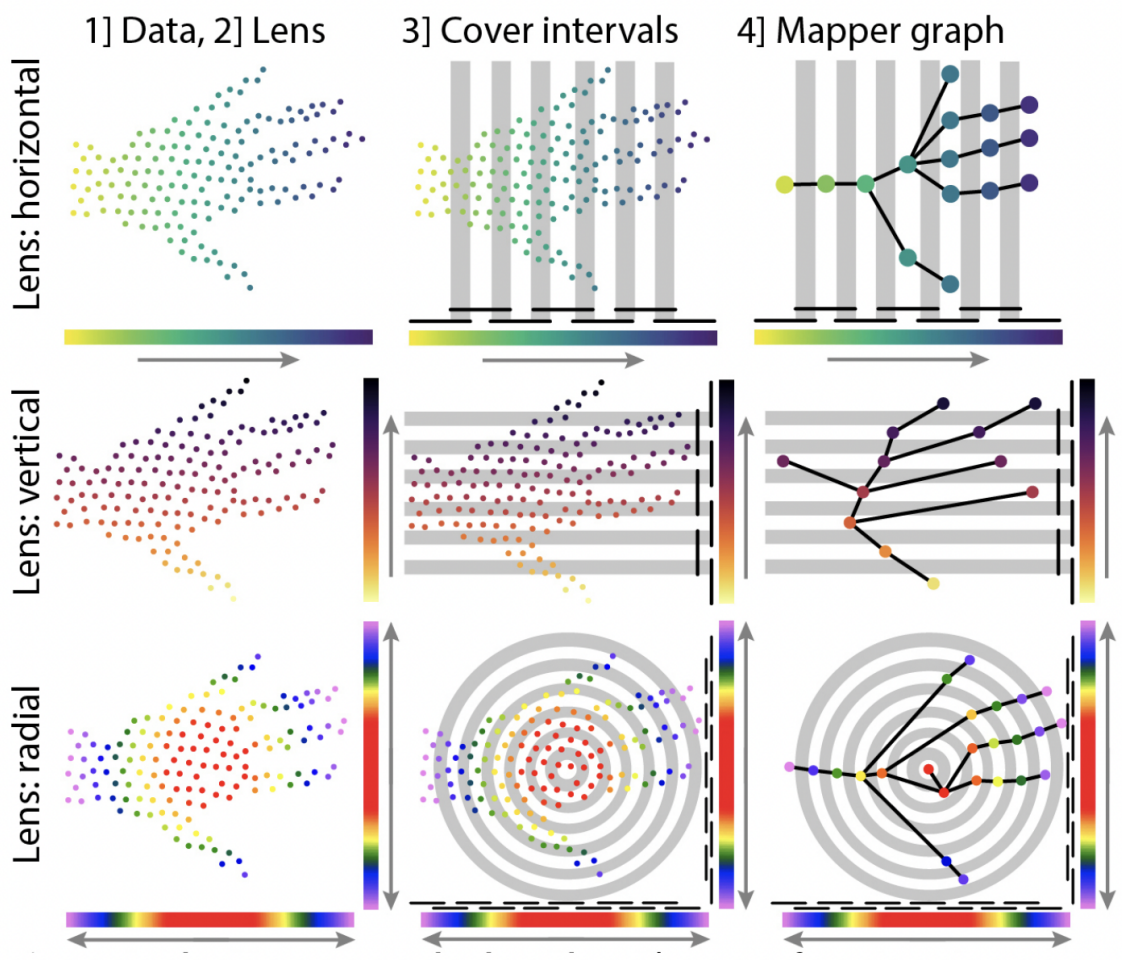

FIGURE 5 – *Three different lenses(filters) on the same point cloud data. The rows show the horizontal, vertical and radial filter selection results, respectively. The image is taken from [26].*

Filter selection is a key point of the Mapper algorithm. In Fig. 5, three different filters are applied to the same point cloud data such as vertical, horizontal, and radial filters. Although the point cloud, in this case, is small with only 2 dimensions, this is a good example of how filters could change the Mapper result. Filter choice depends on the data, complexity of data, and the goal of the Mapper user. A different filter function will generate a graph with a different shape, thus filters allow one to explore the data from different mathematical perspectives. Here is a list of filter functions that I used in my internship work :

- **A column of the dataset :** If we want to visualize the data according to a column in the data we can choose that column as a filter. Using a specific column as a filter causes the layout to separate on the variables of that column. For example, if one considers a dataset of patients that have different types of breast cancer. These types are represented in the dataset in a specific column. The goal is to visualize the data based on different cancer types. Then, that specific column can be used to separate the data points based on it [27]. Another example is that gender information of patients can be used as a filter. If there is two gender in the data, using gender column will put the data points into a two different groups. I used the gender column in my mapper implementation as the first filter.
- **L_p Norm :** The filter function can be chosen based on the L_p norm for each data point. Below, $f_{p,k}$ represents the filter function, $\vec{V}$ denotes the coordinates of each row vector : $\vec{V} =< f_1, f_2, ..., f_s >$ where f_i represents the ith feature(column) and s is total number of features in the dataset. The L_p norm of an individual patient can be calculated as follows :

$$f_{p,k}\left(\vec{V}\right) = \left(\sum_{i=1}^{d} |f_i|^p\right)^{k/p}. \tag{1}$$

  Note that when $p = 2$ and $k = 1$, then the above formula corresponds to the L_2 (i.e., Euclidean) norm of a row vector.
- ***L*-infinity centrality :** for each data point y, it amounts to finding the maximum distance from y to any other point in the dataset. Basically, it assigns each data point to the furthest distance from itself to another data point. Large values of this filter put points that are far away from the center of the dataset. The L-infinity centrality can be implemented with a few lines of code, as shown in Fig. 6 below.
- **Singular Value Decomposition (SVD) :** SVD is a linear dimension reduction technique. Unlike PCA, it does not center the data points before the actual compu-

```
from scipy.spatial.distance import pdist, squareform

pairwise_dist = squareform(pdist(data, 'euclidean'))
L_infinity_filter = np.amax(pairwise_dist, axis = 1)
```

FIGURE 6 – *A code snippet that shows how to calculate the L-infinity centrality.*

tation, meaning that it efficiently works with sparse data. In the present work, I have used the Truncated SVD function from Sklearn [28]. Here, the important part is that one uses the output vector of the Truncated SVD function after using "fit_transform" as the filter vector.

- **Principal Component Analysis (PCA) :** PCA is a well-known dimension reduction technique. The first component of the PCA result is used as a filter. I have used the PCA function from Sklearn [29].

The choice of the filter is an important factor for achieving a good result in Mapper. Instead of using one filter, one can use more filters to separate the data better. Applying multiple filters is a popular choice, as for instance in [12] where the authors have used the L-infinity centrality as well as principal metric singular value decomposition (SVD1), and managed to identify a new subgroup of type-2 diabetes.

3.2 Selecting gain and resolution parameters

Resolution refers to the number of intervals required to cover each filter image ; the output is a natural number. If one increases the resolution, then the number of communities obtained through the Mapper result increases as well. For example, in Fig. 7, the resolution is selected as three since there are three intervals.

The gain is the overlap percentage of the intervals covering each filter image. The range of the gain parameter is $[0, 1]$; it is real number. If one increases the gain, then the connection between communities increases too and, as a result, the number of edges between nodes in the Mapper result increases. Users should decide about these two parameters carefully in order to obtain an efficient topological representation of their data.

3.3 Selecting clustering algorithm

After mapping the data points into corresponding intervals, the data points get clustered in each interval separately. For this phase, any clustering algorithm can be used.

An example of clustering is shown in Fig. 7 (middle panel). In order to obtain good results, I have tried both the "DBSCAN" and the "Agglomerative" clustering methods from Sklearn [30, 31].

3.4 Resulted Mapper visualization

Finally, a simplicial complex is created and it looks like a graph (see Fig. 7, right panel). The vertices/nodes of the graph represents the cluster sets. The edges represent connectivity information of the complex where an edge is added between two nodes whenever two clusters share the same data points.

Coloring the nodes makes the simplicial complex more understandable. The color of a vertex shows the value of the filter function. For instance, in Fig. 7, yellow corresponds to high values and blue to low values. More details about the parameters of the Mapper algorithm and how to choose them wisely are given in the following sections.

A very good visualization output can be achieved by tuning parameters of the Mapper algorithm, such as filters, gain, resolution, clustering algorithm, and hyper-parameters inside the clustering algorithm. To obtain the results presented in this report, I have chosen the following parameter values :

- Gender column as the first filter,
- L-infinity centrality as the second filter,
- Resolution of gender column filter : 2,
- Gain of gender column filter : 0.01,
- Resolution of the L-infinity centrality filter : 19,
- Gain of the L-infinity filter : 0.02,
- Clustering algorithm : DBSCAN with metric correlation, and *min_samples* parameter is 3. The other parameters stayed default.

With this setup, I have obtained the Mapper complex shown in Fig. 7. Firstly, using the gender column as the first filter divided the data points into two group according to patient's genders. Then, L-infinity centrality filter helped us to find subgroups of patients by places them far away from the center.

4 Community Detection

Community detection is the process where unique groups are found from the complete result of the Mapper algorithm. What makes these groups unique is that they share

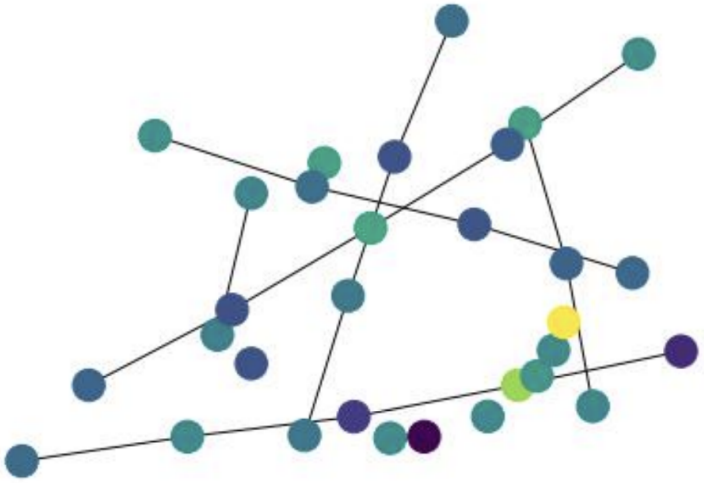

FIGURE 7 – *The mapper complex obtained after applying the Mapper algorithm.*

different characteristics. First, we find the different communities in the Mapper complex, and then we use some statistical tests to identify what unique features represent each community.

4.1 Finding communities

Finding different communities from the result of Mapper amounts to finding the topological features of the simplicial complex. Up to this point, the Mapper algorithm has converted our data into a simplicial complex which is called the Mapper Complex. Topological features of the Mapper complex represent 0- or 1-dimensional topological features such as connected components, up/down branches, and loops. Therefore, topological features of the Mapper complex represent the different communities. So if we extract topological features from the Mapper complex, we can obtain these communities.

I have used the GUDHI library to extract topological features. Connected components and loops are computed with SciPy functions [32], and branches are detected with "Union-Find" and 0-dimensional persistence of the 1-skeleton. Each shape in the Fig. 8 below shows one community colored in yellow.

4.2 Finding characteristics of communities

the different communities contained in the data have been identified, following the procedure explained in the previous section, one finds which features are special in each communities. To do so, one computes the coordinates that best explain a set of nodes com-

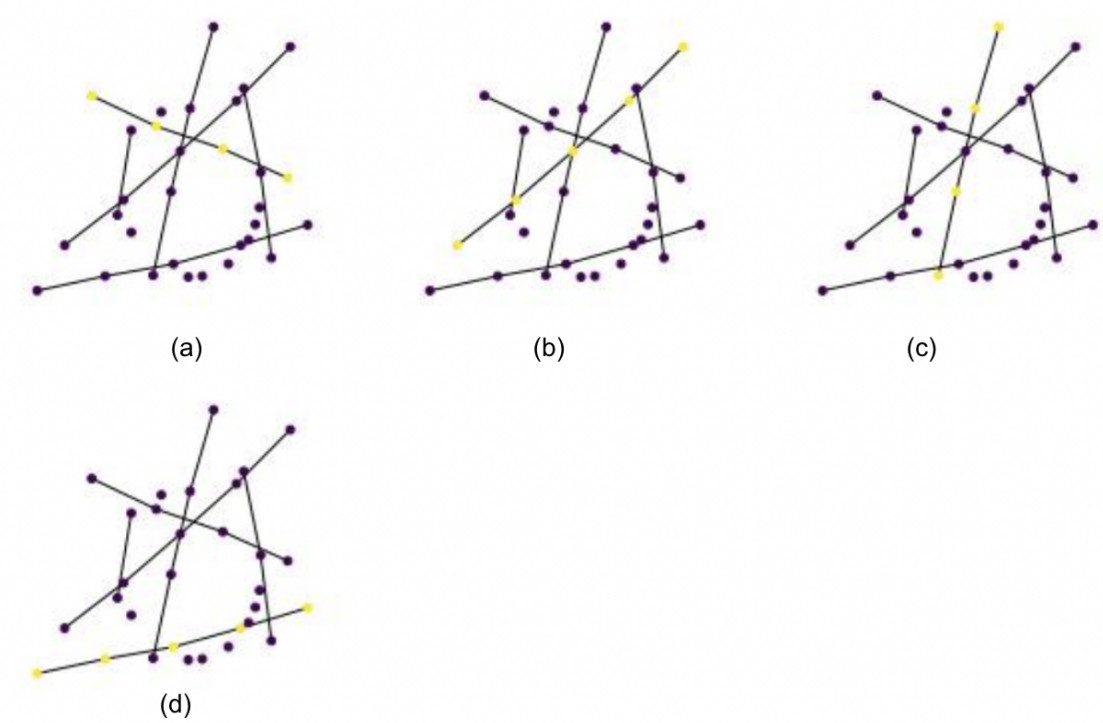

FIGURE 8 – *Different communities found by Mapper are shown in yellow : (a) displays the community 1 ($n = 205$), which consists only of female patients ; (b) is the community 2 ($n = 204$), which consists only of male patients ; (c) and (d) show community 3 ($n = 204$) and community 4 ($n = 77$), respectively, and they both consist of female patients only.*

pared to the rest of the nodes with a Kolmogorov-Smirnov (KS) test. For each community, all the features in the dataset are considered in the KS test.

If $\vec{V}$ denotes the row vector of each patient, then the coordinates of each row vector are given by : $\vec{V} =< f_1, f_2, ..., f_s >$ where f_i represents the ith feature (column) and s is total number of features/columns in the dataset. If one assumes that C_1,C_2,..., C_t are communities, then each represents a different topological feature of the Mapper complex.

Suppose we want to find traits that uniquely define C_1. We write

$$C_1 =< p_1, p_2, ..., p_k >, \tag{2}$$

where p_k is the kth data point that creates the first community, C_1 Then, to find the traits of C_1 we proceed as follows.

First, we divide the dataset into two groups. Data points in the first group form C_1, and the rest of the data points except C_1 form another group. Then, for each feature in the dataset, we examine two different distributions. The first distribution has feature

values of C_1, the second distribution has the feature values of the rest. Recall that these feature values belong to just one feature(column) of the whole dataset. Then, these two distributions are tested using a Kolmogorov-Smirnov(KS) test. The output of the KS test is a p-value that shows the similarity between the two distributions. A larger p-value indicates that the two distributions are statistically similar. If a p-value of a feature is smaller than 0.05, then that feature is a unique characteristic of C_1 and we can say that feature represents that community.

From a practical viewpoint, I have computed p-values for each feature in one community and I only considered p-values less than 0.05. Then, I have sorted them in ascending order in a list, the first element of the list representing the most representative feature of that community.

The pseudo-code of how to find unique features of C_1 is written below : In this algorithm, KS denotes the Kolmogorov-Smirnov test. I used the two-sample Kolmogorov-Smirnov test from the Scipy-stats library [33] to test the importance of a feature. The above algorithm is done for each community <C_1, C_2, ... , C_t>. In the end, I calculated statistically important features for each community (with p-value less than 0.05).

Algorithm 1 *How to find unique features in a community.*

$X \leftarrow$ data points that creates C_1
$Y \leftarrow$ rest of the data points except X
for $feature(f) = 1, 2, \ldots s$ **do**
    $group1 \leftarrow X_{f_1}$ #data points in C1 belong to a feature f_1
    $group2 \leftarrow Y_{f_1}$ #rest of the data points belong to feature f_1
    $Pvalue \leftarrow KS(group1, group2)$

    $UniqueFeatures = [\,]$ #Unique features that represents C_1
    if $Pvalue < 0.05$ **then**
        $UniqueFeatures.append(f_1)$
    end if
end for
$UniqueFeatures = sort(UniqueFeatures)$ #Unique features of C_1 in ascending order

5 Results

Each community in the Mapper complex represents a different group in the data. There are four different communities detected by Mapper which are shown in Fig. 8. We see three different groups composed by female patients only, and one group of only male patients. The genders among communities are heterogeneous since the gender column is used as the first filter function for Mapper. Female groups are named as F_1 ($n = 205$), F_2 ($n = 204$), F_3 ($n = 77$) and the male group is called M_1 ($n = 205$). I have analyzed the cognitive, emotional, and motor skills in each group, and identified which features make these groups different. Clinical features specific to groups are listed below where p-values are in ascending order from top to bottom i.e. first row in the tables is the most representative feature of that community.

Clinical features	**Category**	**p-value**	$\mathbf{F_1}$	$\mathbf{M_1}$	$\mathbf{F_2}$	$\mathbf{F_3}$
Gender	Other	< 0.05	✓	✓	✓	✓
Strength (Grip Strength Dynamometry)	Motor	< 0.05	✓	✓	✓	✓
Language/Vocabulary Comprehension	Cognition	< 0.05	✓			
Dexterity (9-hole Pegboard)	Motor	< 0.05	✓	✓	✓	✓
Endurance(2 minute walk test)	Motor	< 0.05	✓	✓	✓	✓
Spatial Orientation	Cognition	0.001	✓	✓		
Fluid Intelligence	Cognition	0.002	✓			
Episodic Memory	Cognition	0.003	✓	✓	✓	✓
Stress and Self Efficacy	Cognition	0.005	✓	✓		✓
Language/Reading Decoding	Cognition	0.006	✓			
Social Relationships - Loneliness	Emotion	0.008	✓	✓		
Processing Speed	Cognition	0.019	✓	✓	✓	
Sustained Attention	Cognition	0.035	✓		✓	

TABLE 1 – **(A) Clinical features for $\mathbf{F_1}$ ($\boldsymbol{n = 205}$, only women)**.

Among the four groups, gender and strength (grip strength dynamometry) are the first two common features that define these communities with the smallest p-values ($<$ 0.05). The grip strength dynamometry test is adopted from the *American Society of Hand Therapy*, where the participants sit on a chair, bend their elbows at 90 degrees, and are asked to squeeze to dynamometer as hard as they can with their right and left hands. The result of the test provides a digital score of force in pounds. This motor skill shows how

Clinical features	Category	p-value	F_1	M_1	F_2	F_3
Gender	Other	< 0.05	✓	✓	✓	✓
Strength (Grip Strength Dynamometry)	Motor	< 0.05	✓	✓	✓	✓
Dexterity (9-hole Pegboard)	Motor	< 0.05	✓	✓	✓	✓
Endurance (2 minute walk test)	Motor	< 0.05	✓	✓	✓	✓
Stress and Self Efficacy	Emotion	< 0.05	✓	✓		✓
Episodic Memory	Cognition	< 0.05	✓	✓	✓	✓
Negative Affect - Sadness	Emotion	0.005	✓			
Spatial Orientation	Cognition	0.001	✓	✓		
Processing Speed	Cognition	0.006	✓	✓	✓	
Social Relationships -Loneliness	Emotion	0.013	✓			
Verbal Episodic Memory	Cognition	0.013		✓	✓	

TABLE 2 – **(B) Clinical features for M_1 ($n = 205$, only men)**.

strong the person is. The average value of strength in the women groups is 92.31, 87.54, and 84.23 in F_1, F_2 and F_3, respectively. The average strength in the male group is larger than in the female groups, with a value of 122.28. In general, physical power is greater in men than women, which is why strength became a key feature to represent these groups.

There are also three other features shared by all of the groups, namely dexterity (9-hole Pegboard), endurance (2 minutes walk test), and episodic memory. Dexterity is a test where participants are asked to put and remove 9 plastic pegs into a pegboard with each

Clinical features	Category	p-value	F_1	M_1	F_2	F_3
Strength (Grip Strength Dynamometry)	Motor	< 0.05	✓	✓	✓	✓
Gender	Other	< 0.05	✓	✓	✓	✓
Endurance (2 minute walk test)	Motor	< 0.05	✓	✓	✓	✓
Episodic Memory	Cognition	< 0.05	✓	✓	✓	✓
Executive Function/ Inhibition	Cognition	0.004			✓	
Verbal Episodic Memory	Cognition	0.014		✓	✓	
Processing Speed	Cognition	0.032	✓	✓	✓	
Negative Affect - Anger	Emotion	0.037			✓	
Dexterity (9-hole Pegboard)	Motor	0.042	✓	✓	✓	✓
Sustained Attention	Cognition	0.046	✓	✓		✓

TABLE 3 – **(C) Clinical features for F_2 ($n = 205$, only women).**

Clinical features	Category	p-value	F_1	M_1	F_2	F_3
Gender	Other	< 0.05	✓	✓	✓	✓
Strength (Grip Strength Dynamometry)	Motor	< 0.05	✓	✓	✓	✓
Episodic Memory	Cognition	< 0.05	✓	✓	✓	✓
Dexterity (9-hole Pegboard)	Motor	0.002	✓	✓	✓	✓
Endurance (2 minute walk test)	Motor	0.003	✓	✓	✓	✓
Stress and Self Efficacy	Emotion	0.023	✓	✓		✓

TABLE 4 – **(D) Clinical features for F_3 ($n = 77$, only women).**

hand. The score is the time taken by the participants (3-85 years old) to finish the task. An endurance test is used to test the sub-maximal cardiovascular endurance. Participants walk a 50-foot long lane (back and forth) in 2 minutes and the score is the distance they were able to achieve. On the other hand, episodic memory measures how well one can remember previous experiences within the context of location, time, or emotions. Participants are asked to recall the sequence of objects that are in a specific order shown on the computer screen. If they can remember correctly the order of two objects they get one point. All these three tests resulted to be common features to represent the four groups. Here, it is normal to have common features among groups, yet the important part of TDA analysis is to find specific characteristics that are not shared, which we analyze group by group below.

Group F_1 There are three variables that are specific to this group : (i) Language/vocabulary comprehension, (ii) fluid intelligence and (iii) language/reading decoding. Language/vocabulary comprehension tests the ability of a person to correctly match words with images. In the test, an audio record of a word is given to the person and she/he picks an image out of four images that are most related to the word. The mean value for participants in F_1 in this test is 105.52. Other groups' average scores are 108.86, 106.01, and 105.39 respectively. The mean values are similar but this feature becomes a discriminative property for F_1.

Fluid intelligence aims to capture how successful a given person is at thinking and finding reasons in an abstract way and solving problems. This female group was found to be better in this test compared to other patients. Language/reading decoding measures the ability of the patient to accurately read and pronounce the words in both English and Spanish. The F_1 group has a disjunctive characteristic of this feature with a mean value

of 105.5. There is no feature that only defines female groups but not males.

Group F_2 This female group is detected by the variable "negative affect-anger". Anger is described as the attitude of a person about cynicism, hostility and frustrating experiences. People in F_2 are linked to anger problems. The other feature that is specific to F_2 is "executive function/inhibition". Participants' attention and inhibitory control are tested. Our results indicate that the F_2 group is characterized by attention skills.

Group F_3 This group has no specific feature that only belongs to itself. Besides the common features of the four groups, F_3 is linked to stress and self-efficacy. However, this variable is also shared by F_1 and M_1. I interpreted this group as an average female group for which there is no diverse trait.

Group M_1 Interestingly, we have one male group in the resulted communities. Men in this group are linked with two specific emotional categories : Sadness and loneliness. These features do not appear in female groups. Surprisingly, men in the dataset seem to have issues with sadness and loneliness. It makes sense that loneliness appears with sadness because they are related emotions. Patients in M_1 share the verbal episodic memory with patients in F_2, and spatial orientation feature with patients in F_1.

Conclusion

In this internship, I have used the HCB dataset that includes mostly cognitive, motor, and emotional characteristics. I have used two topological data analysis methods to analyze these data. The first one is using the Sparse-Rips complex to convert brain connectivity matrices of patients into scalar features. The second method is the Mapper algorithm. Mapper converts patients' data into a Mapper complex with some filter functions, gain, and resolution parameters. Filters are scalar vectors that are used to map the data points and divide them into different communities. There are many filter options such as L_p norm, L-infinity centrality, first or second components of SVD, or just a column in the data. I have used the gender column and the L-infinity centrality as filter functions. Topological features of the Mapper complex have resulted in the identification of the different communities. It is a hard task to tune the parameters in the Mapper to have a meaningful result. I have used the GUDHI library for the topological functions. After finding the communities, I have used Kolmogorov-Smirnov(KS) test to statistically prove representative features in each community. If the p-value of the KS test was less than 0.05 for a feature, I have considered it as a specific variable to that community. Then I have made the analysis of different groups and identified the specific characteristics of each group. The male group turned out to be linked with loneliness and sadness, while female groups seemed more linked to cognitive skills.

During this internship, I have learned topological data analysis, how to apply it to a complex dataset and interpret the result. TDA is a powerful tool for a data scientist to explore the data in a unique way where classical data analysis methods may fail. I have learned what insights topology can give to a data scientist. Topology was a black box for me but after learning how to apply it, I found it very powerful and particularly rewarding.

Appendix

The correlation matrix of the data columns is shown in Fig. 9 below.

Comments on the correlation matrix :

- If a value is high in a cell in the matrix, it means that these two features are highly correlated. On the other hand, if the value is low, that means these features are independent and do not affect each other.
- If a value is positive, that means two variables are positively correlated. If a value is negative, that means two variables are negatively correlated. A negative correlation represents the relationship between two variables that if one variable increases as the other decreases, and vice versa. For example, as seen in the matrix studied in the present work, gender and strength are negatively correlated.
- The matrix is symmetric. The diagonal is always 1 because the correlation between each feature and itself is necessarily equal to 1.

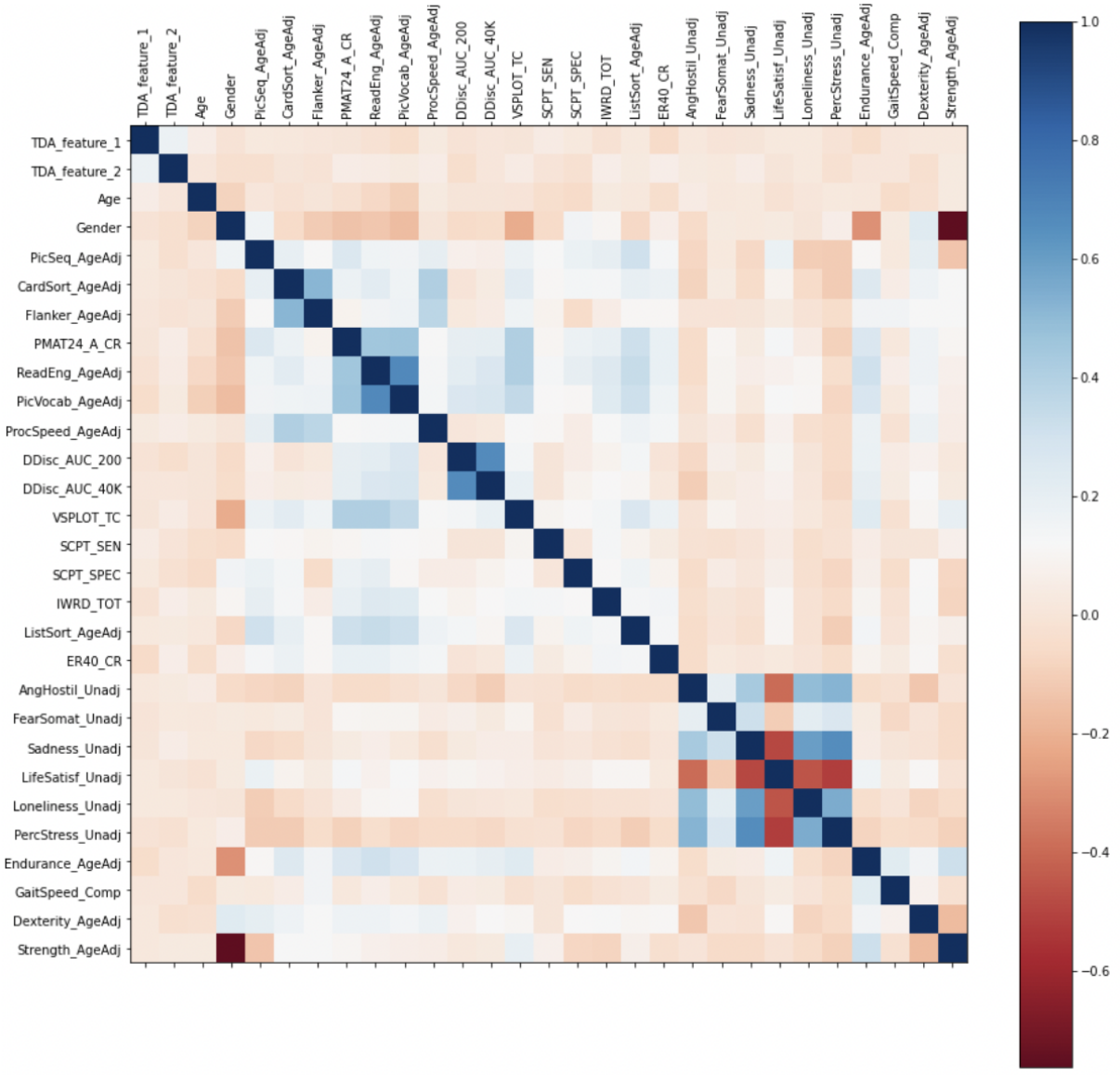

FIGURE 9 – *Correlation matrix of data columns. The labels in the image represent the columns.*

Bibliographie

[1] Leonhard Euler. Solutio problematis ad geometriam situs pertinentis. *Commentarii Academiae Scientiarum Petropolitanae*, pages 128–140, 1741.

[2] Miroslav Kramár, Arnaud Goullet, Lou Kondic, and Konstantin Mischaikow. Persistence of force networks in compressed granular media. *Physical Review E*, 87(4) :042207, 2013.

[3] Takenobu Nakamura, Yasuaki Hiraoka, Akihiko Hirata, Emerson G Escolar, and Yasumasa Nishiura. Persistent homology and many-body atomic structure for medium-range order in the glass. *Nanotechnology*, 26(30) :304001, 2015.

[4] Primoz Skraba, Maks Ovsjanikov, Frederic Chazal, and Leonidas Guibas. Persistence-based segmentation of deformable shapes. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pages 45–52. IEEE, 2010.

[5] Katharine Turner, Sayan Mukherjee, and Doug M Boyer. Persistent homology transform for modeling shapes and surfaces. *Information and Inference : A Journal of the IMA*, 3(4) :310–344, 2014.

[6] Talha Qaiser, Yee-Wah Tsang, Daiki Taniyama, Naoya Sakamoto, Kazuaki Nakane, David Epstein, and Nasir Rajpoot. Fast and accurate tumor segmentation of histology images using persistent homology and deep convolutional features. *Medical Image Analysis*, 55 :1–14, 2019.

[7] Bastian Rieck, Tristan Yates, Christian Bock, Karsten Borgwardt, Guy Wolf, Nicholas Turk-Browne, and Smita Krishnaswamy. Uncovering the topology of time-varying fmri data using cubical persistence. *Advances in Neural Information Processing Systems*, 33 :6900–6912, 2020.

[8] Firas A Khasawneh and Elizabeth Munch. Chatter detection in turning using persistent homology. *Mechanical Systems and Signal Processing*, 70 :527–541, 2016.

[9] Lee M Seversky, Shelby Davis, and Matthew Berger. On time-series topological data analysis : New data and opportunities. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 59–67, 2016.

[10] Yuhei Umeda. Time series classification via topological data analysis. *Information and Media Technologies*, 12 :228–239, 2017.

[11] Meryll Dindin, Yuhei Umeda, and Frederic Chazal. Topological data analysis for arrhythmia detection through modular neural networks. In *Canadian Conference on Artificial Intelligence*, pages 177–188. Springer, 2020.

[12] Li Li, Wei-Yi Cheng, Benjamin S Glicksberg, Omri Gottesman, Ronald Tamler, Rong Chen, Erwin P Bottinger, and Joel T Dudley. Identification of type 2 diabetes subgroups through topological analysis of patient similarity. *Science Translational Medicine*, 7(311) :311ra174–311ra174, 2015.

[13] Yuan Yao, Jian Sun, Xuhui Huang, Gregory R Bowman, Gurjeet Singh, Michael Lesnick, Leonidas J Guibas, Vijay S Pande, and Gunnar Carlsson. Topological methods for exploring low-density states in biomolecular folding pathways. *The Journal of Chemical Physics*, 130(14) :04B614, 2009.

[14] Mathieu Carrière and Raúl Rabadán. Topological data analysis of single-cell hi-c contact maps. In *Topological Data Analysis*, pages 147–162. Springer, 2020.

[15] Yongjin Lee, Senja D Barthel, Paweł Dłotko, S Mohamad Moosavi, Kathryn Hess, and Berend Smit. Quantifying similarity of pore-geometry in nanoporous materials. *Nature Communications*, 8(1) :1–8, 2017.

[16] GUDHI, `https://gudhi.inria.fr/`.

[17] Scikit-TDA, `https://scikit-tda.org/`.

[18] Giotto-da, `https://giotto-ai.github.io/gtda-docs/0.5.1/library.html`.

[19] Dataset-1, `https://tinyurl.com/2vkmp2rb`.

[20] Dataset-2, `https://tinyurl.com/523ve8jr`.

[21] Sklearn Categorical Encoders, `https://contrib.scikit-learn.org/category_encoders/`.

[22] SparseRipsPersistence, `https://tinyurl.com/2p86tcfa`.

[23] Persistence Entropy, `https://tinyurl.com/477p69h8`.

[24] Gurjeet Singh, Facundo Mémoli, Gunnar E Carlsson, et al. Topological methods for the analysis of high dimensional data sets and 3d object recognition. *PBG@ Eurographics*, 2, 2007.

[25] Mapper Image, `https://www.youtube.com/watch?v=NjcLSviRP5E&t=931s`.

[26] Mapper Image-2, `https://link.springer.com/article/10.1007/s41468-022-00090-w`.

[27] Pek Y Lum, Gurjeet Singh, Alan Lehman, Tigran Ishkanov, Mikael Vejdemo-Johansson, Muthu Alagappan, John Carlsson, and Gunnar Carlsson. Extracting insights from the shape of complex data using topology. *Scientific reports*, 3(1) :1–8, 2013.

[28] Truncated SVD, https://tinyurl.com/26rdaz2h.

[29] PCA in Sklearn, https://tinyurl.com/mvvuezds.

[30] DBSCAN(.), https://tinyurl.com/4xrsyz7n.

[31] Agglomerative Clustering, https://tinyurl.com/mr2u2mut.

[32] SciPy, https://scipy.org/.

[33] Kolmogorov-Smirnov test, https://tinyurl.com/kn5c3ztp.