# TDA technologies on fMRI

## what topology can tell us about data

Bachelor of Science

by

Marta Visetti

Computer science

Université de Fribourg

Under the supervision of

Prof. Bastian Rieck

Université de Fribourg

AIDOS Lab

UNI FR

UNIVERSITÉ DE FRIBOURG
UNIVERSITÄT FREIBURG

# Acknowledgements

Template credits: Biplab Mahato.

# Abstract

Functional magnetic resonance imaging (fMRI) is widely used to study neurodegenerative diseases, nonetheless extracting discriminative features for automated classification remains challenging. This thesis evaluates the use of Topological Data Analysis (TDA), combined with functional connectivity features, to distinguish neurodegenerative patients from healthy controls. Persistent homology is applied to functional connectivity matrices derived from ENIGMA resting-state fMRI data to extract topological descriptors, which are evaluated using multiple machine learning models with and without Principal Component Analysis (PCA). Across all experimental settings, classification performance consistently results at a non-deterministic level, with accuracy and ROC_AUC values close to 0.5. This indicates that neither model choice nor PCA is the primary limiting factor, suggesting that global topological summaries of functional connectivity may be insufficient for binary disease classification.

# Contents

# Introduction

Neurodegenerative diseases, of which Alzheimer's and Parkinson's disease are the most common, constitute a major public health challenge. [29] This is particularly evident with the aging of the population where more and more people are affected by these diseases. Early detection and accurate characterization of these disorders are crucial for improving clinical outcomes and for advancing the understanding of their underlying mechanisms. Resting-state functional magnetic resonance imaging (rs-fMRI) has emerged as a powerful non-invasive tool for studying large-scale brain organization, allowing the analysis of functional connectivity patterns across distributed brain regions, the regions of interest. Extracting features that are both robust and discriminative from fMRIs for automated disease classification remains challenging. In recent years, Topological Data Analysis (TDA) has been proposed as a complementary framework for analyzing complex, high-dimensional data. By focusing on the shape of data across multiple scales, TDA offers descriptors that are theoretically robust to noise and invariant to certain transformations. Instead of the classical data approach we aim for a study of the underlying shape of data and then analyze the results statistically.

Motivated by these properties, this thesis investigates the applicability of TDA to fMRI data in the context of neurodegenerative disease classification. In particular, persistent homology is applied to functional connectivity matrices derived from the ENIGMA dataset to extract topological descriptors, including Betti curves, persistence landscapes, persistence silhouettes, and persistence images. These features are evaluated using gradient-boosting-based classification models, both with and without dimensionality reduction via Principal Component Analysis (PCA).

This thesis aims to assess whether TDA-based representations of functional connectivity provide discriminative information for distinguishing neurodegenerative patients from healthy controls. On this matter, it then examines the impact of PCA on the performance of classifiers trained on TDA-derived features. This thesis focuses on an evaluation of an existing analysis pipeline and its adaptations.

The remainder of this thesis is organized as follows. *Part I* states some prior knowledge. *Chapter 1* introduces the theoretical background of Topological Data Analysis and persistent homology. *Chapter 2* quickly describes some fundamentals of neurological data sets.

*Part II* is implementation driven. *Chapter 3* focus on the preexisting pipeline describing its functioning. *Chapter 4* defines and describe the modifications made for the new approach. *Part III* reflects on the results obtained. *Chapter 5* discusses the implications of the findings and outlines directions for future research.

# Part I

# Background and theory

# Chapter 1

# Topological Data Analysis

## 1.1 Introduction

The main focus of this thesis is on the benefits one gets from applying topological data analysis to the research field. This method introduces an innovative approach to studying the shape of data[22] and do our analysis related to this information. Starting from the idea that data are sampled from a geometric object; we want to find this intrinsic object back to study the data under a geometrical perspective. [10] The emerging area of topological data analysis focuses on the recovery of the lost topology of this underlying space [43]. For the reader to understand the usage of Topology Data Analysis (TDA) on the data, we first need to clarify some basic topological concepts and other algebraic tools that will be used in the work. This chapter guides the reader through them.

## 1.2 Homology

Homology is a fundamental tool of algebraic topology that is used to assign invariants to topological spaces; the latter maintain essential information about their original structure [25]. For the active usage in machine learning, homology is a less resource costly and easier to compute tool than others like homotopy, but can still be very helpful in retrieving information and classifying the data.

**Definition 1.** ***Homology groups*** *are topological invariants of topological spaces; defined using chains, cycles, and boundaries. [25]*

**Definition 2.** *An* ***Homology class*** *is one element of a homology group; homology classes represent individual topological features.*

## 1.3 Holes and Betti numbers

### 1.3.1 Topological holes

In this thesis, the homological components taken into consideration and studied are of dimension 0 and 1 in homology, which correspond to connected components and tunnels. Intuitively, connected components represent sets of mutually connected points, while tunnels correspond to loops in a graph or, more generally, to one-dimensional topological holes.

### 1.3.2 Betti numbers

The $d^{th}$ Betti number counts the number of d-dimensional holes. Objects can be classified and differentiated in algebraic topology according to their Betti numbers. For each degree $k$, the $k^{th}$ Betti number $\beta_k$ is defined as the rank of $H_k(\Sigma)$ and it counts the number of independent k-cycles which do not represent the boundary of any collection of simplices of $\Sigma$. When studying low dimensional spaces, we mainly focus on the connected components $\beta_0$, tunnels and their holes $\beta_1$ and the shell surrounding voids or cavities $\beta_2$. [21]
Betti numbers can explain the principle behind the famous statement in topology that

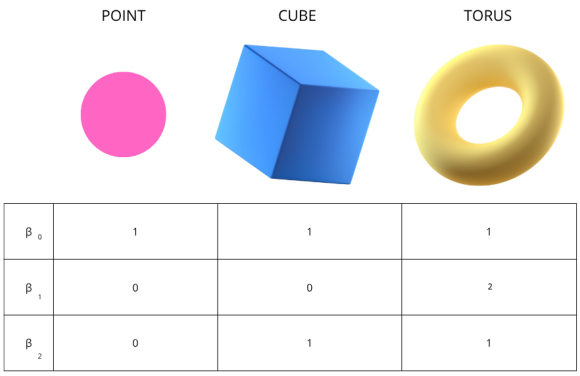



Figure 1.1: Examples of topological spaces with their corresponding Betti numbers. Figure created with Canva tool.

a torus and a mug are the same shape [33]. This is supported by the fact that they share the same values of Betti numbers and are hence in the same topological class. They share the same Betti numbers: one connected component ($\beta_0 = 1$), two independent tunnels ($\beta_1 = 2$), and one enclosed void ($\beta_2 = 1$).

Figure 1.2: Image of a mug morphing into a torus

## 1.4 Simplicial homology

It is possible to construct topological invariants of different objects using simplicial complexes. This is particularly interesting as we are able to represent the topological space by using simple pieces [15].
The first element to consider in simplicial homology are **simplices** as they are the building blocks of simplicial complexes.

**Definition 3.** *An **n-simplex** is a geometric object with (n+1) vertices which lives in an n-dimensional space; it is a generalization of the notion of a triangle or tetrahedron to arbitrary dimensions. [28]*

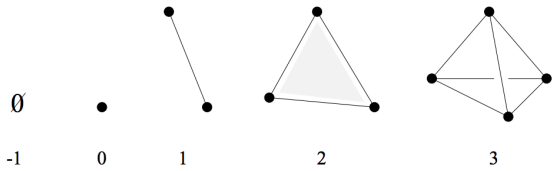



Figure 1.3: Visualization of simplices on dimension going from -1 to 3. From dimension 0 to 3 one can observe in order: a vertex, an edge, a triangle and a tetrahedron

A Simplex, being a topological element, can be rotated, translated, dilated and streched without changing its identity; this does not always hold when crushing it as we might create a lower dimension simplex.

**Definition 4.** *A **face** of a simplex is a simplex generated by a subset of the original simplex.*

**Definition 5.** *A **simplicial complex** $K$ is a collection of simplices such that if $K$ contains a simplex, it must contain all its faces and if two simplices in $K$ intersect, then their intersection is a face of each of them. In general a d-simplex is the convex hull of d+1 points.It is a data structure able to represent the topological space.*

**Definition 6.** *The **dimension** of a simplicial complex is the maximum dimension of the simplices composing it. [25]*

**Definition 7.** *A **triangulation** of a topological space $X$ is given by a simplicial complex $K$ together with an homeomorphism between $X$ and $|K|$ .[15]*

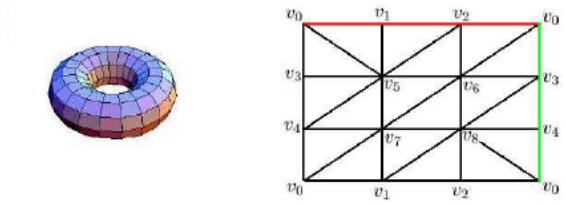



Figure 1.4: A triangulation $T$ of a torus. Image from [23]

To build simplicial complexes from cloud data, one can pass by the creation of a Čech complex, in which a simplex is included whenever the corresponding fixed-radius balls centered at the data points have a nonempty common intersection [24].

## 1.5 Distance-based complexes

The motivation to use Vietoris–Rips and Čech complexes in the context of data analysis is that these complexes can recover topological features of an unknown sample space underlying the data, since they can be used to reconstruct some simplicial complexes from the data. [2] The filtration method applied uses the distance between points to create clusters. This choice comes from the fact that small distances between data points often reflect notions of similarity between data points [10].

### 1.5.1 Čech complexes

Čech complexes are build according to the overlap of closed balls having the points of the topological space as centers. The variable one can tune during the creation of these complexes is the radius of the closed balls, $r \geq 0$.

**Definition 8.** *A **closed ball** [2] with radius $r$ and center $x$ in a metric space $(X, d)$, $d$ being a distance, is denoted as*

$$B(x, r) := y \in X | d(x, y) \leq r$$

**Definition 9.** *A **Čech complex** can be defined as*

$$\check{C}ech(X, X'; r) = \{finite \sigma \subseteq X | \bigcap_{X \in \sigma} BX'(x, r) \neq \emptyset\}$$

*that is the finite collection of segments generated by the intersections of balls of radius $r$.*

When a ball intersect only with another one a segment is created, but since 3 balls of 3 points of the space intersect mutually with a triple point of intersection, we start to have a triangulation; the more points intersect mutually creating an high level point of intersection, the higher is the degree of the simplex created [1] [5].

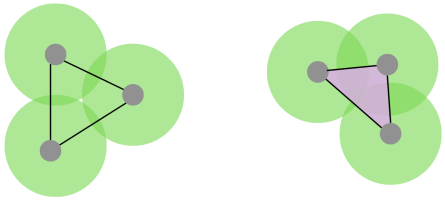


Figure 1.5: Example of the creation of a simplicial complex between 3 points using Čech rules. In the figure on the left there isn't a common point of intersection between the points, while in the second image this point is present, hence we create a triangle. Image created with the help of Canva

### 1.5.2 Vietoris-Rips complexes

**Definition 10.** *Given a metric space $X$ and a distance threshold $\epsilon > 0$ , the **Vietoris–Rips simplicial complex** has as its simplices the finite subsets of $X$ of diameter less than $\epsilon$ [1].*

In contrast to Čech complexes, Vietoris Rips complexes create high dimensional simplices according to the mutually connecting points, no matter whether they have a common point of intersection. The connection between two points is established when the two closed balls built around them collide, hence the two points find themselves at a distance inferior to the diameter of the ball.

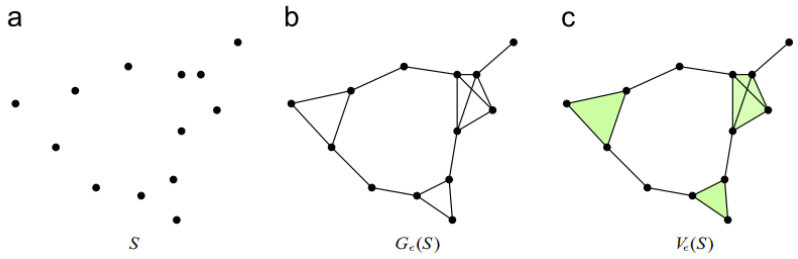



Figure 1.6: Construction of the Vietoris-Rips complex. Our input (a) is a set of points S. Section 3.2 describes the first phase, the geometric process of going from (a) to a neighborhood graph (b) then expanding from the graph (b) to the Vietoris-Rips complex (c), image from [43]

### 1.5.3 Filtration of Vietoris-Rips complexes

The first operation performed on the raw data is filtration. In our case ,filtration is fundamental to go from a point cloud to the simplicial complex that we want to study.

To this end, we construct a topological space using information attached to the set of points composing the data[10].
Since Vietoris-Rips complexes tend to be easier to compute than Čech complexes, we opt to work with them. This is also the reason why they are often used in TDA. In our case, this is the key step to reveal the underlying geometrical structure from the data.
The filtration is performed by increasing the value of the distance threshold $\epsilon$ regulating

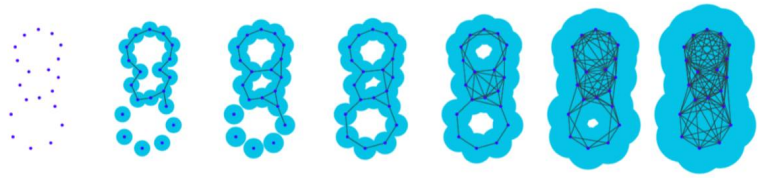


Figure 1.7: A filtration of a simplicial complex $\Sigma$, image from [9]

the creation Vietoris-Rips complexes.

## 1.6 Persistent homology, or topological persistence

Starting from the filtration of the simplicial complex, one can study its persistence. That includes the evolution of homology groups across the filtration, tracking the appearance and disappearance of topological features as the scale parameter varies [16]. This information is encoded in **persistence module**, mathematical structures consisting of vectors spaces and linear maps which summarizes how homological features are created and destroyed throughout the filtration. This procedure is quite interesting as we are adding a new variable to the analysis [37]. This approach is relevant because it yields topological features that are robust to noise, it can help classify the interesting "holes" of a shape and allows a multi-scale approach to shape description [21] while comparing it on a time scale. Persistent homology tracks the creation and destruction of topological features present in the Vietoris-Rips complexes as the value of the distance threshold $\epsilon$ varies increasingly. This method provides some insights about the topological features that are quite robust[1]. Persistent homology describes the changes in homology that occur to an object which evolves with respect to a parameter [41]; in our study the parameter will be $\epsilon$ as described in the previous section and the focus is on the birth and death of holes. The strength behind persistent homology is that it doesn't restrict itself to observe homology groups at a single value of $\epsilon$, which might be not relevant; it rather observe the overall evolution of the homology groups around a big interval of $\epsilon$ revealing the sections of interest.
Nevertheless, there are some downgrades, first of all the computational costs that are high [40]: the time complexity of the algorithm is high, with the standard algorithm taking

cubic running time in regard to the complex size, the memory consumption is large, and the focus of several applications is in data of higher dimension.
Persistent homology uses **persistence pairs** to be visualized. The latter define the birth and death of homology classes as the parameter grows. The persistence is the difference between the death, a class becomes trivial, and birth, a class comes into being[40]. Information about persistence pairs are stored in **persistence diagrams** having as coordinates the persistence pairs of homology classes, each represented by a point. They can later be visualized with many tools which vectorize the persistence diagrams, such as barcodes, persistence landscapes, persistence images[12]. This representation allow the application of traditional statistics as they are sets of functions [8]. With the results one can distinct noise (short living holes) and actual features (recognized by persistence pairs with greater delta between birth and death).

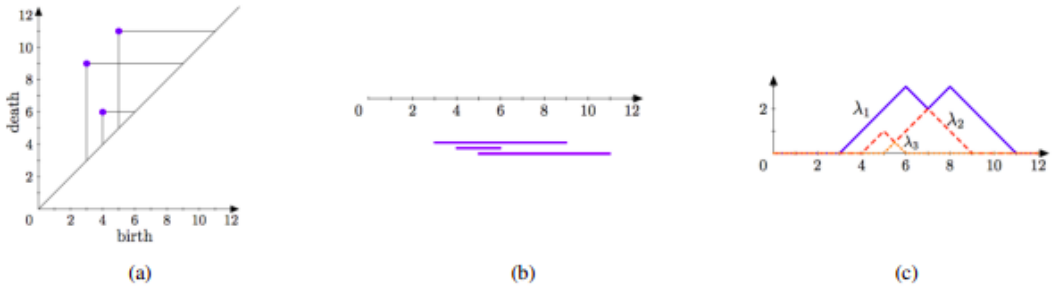



Figure 1.8: Various visualizations of the persistence pairs in degree 1 of the filtration depicted in Figure 1.4: persistence diagram (a), barcode (b) and persistence landscape (c); image from [9]

### 1.6.1 Persistence landscapes

Firstly introduced by Bubenik [9], the main technical advantage of persistence landscapes is that they are functional representation of persistence diagrams; this makes calculations with them are much faster than the corresponding calculations with barcodes or persistence diagrams.
Starting from a persistence diagram, the creation of a persistence landscape is obtained by the mapping of persistence pairs to triangular functions; once these functions are created, the $k$-th largest value of them is taken at each point.

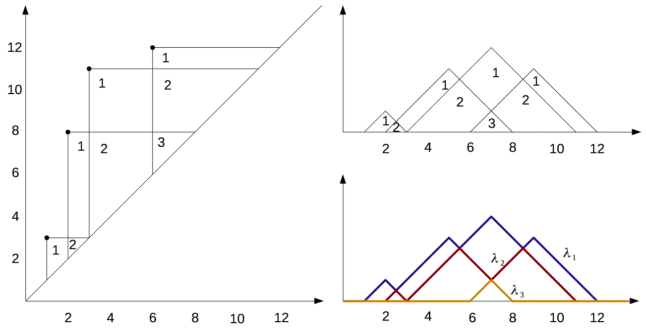


Figure 1.9: The persistence diagram (left) is tilted, so that the diagonal becomes the new horizontal axis (top right); peaks of functions created by points of the same homology dimension are concatenated together to return a sequence of functions (bottom right). Image from [35]

The value on the Y-axis represents the longevity of the homology feature, the X-axis correspond to the filtration parameter.

### 1.6.2 Persistence silhouettes

Persistence silhouettes are taking the values of persistence diagrams and returning a continuous real-valued function, as persistence landscapes do.
Persistent silhouettes are power weighted, meaning that according to a variable value, the function takes into account the most persistent points. [11]. This value $p$ defines whether the results will be dominated by the low persistent pairs or the high persistent ones.

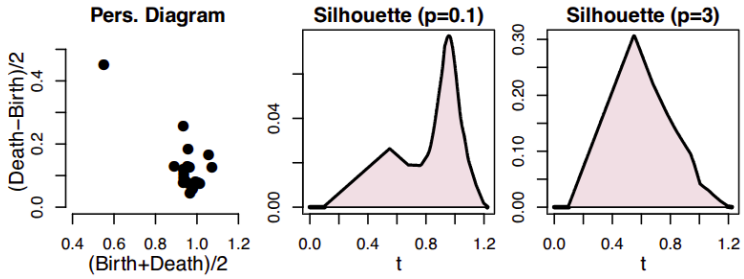



Figure 1.10: An example of power-weighted silhouettes for different choices of p. The axes are on different scales. Image from [11]

### 1.6.3 Betti curves

A Betti curve describes the evolution of topological features across a filtration by plotting the Betti number $\beta_k(\alpha)$ as a function of the filtration parameter $\alpha$. For a fixed dimension $k$, $\beta_k(\alpha)$ counts the number of $k$-dimensional homological features (e.g., connected components for $k = 0$, loops for $k = 1$) present at filtration value $\alpha$. Betti curves provide a

compact summary of persistent homology and can be derived from persistence diagrams by counting the number of intervals alive at each filtration value [39, 30].

### 1.6.4 Persistent images

Persistent images are a finite-dimensional vector representation of persistence diagrams. The first step to create a persistence image is to map a persistence diagram $B$ to a persistence surface $\rho_B$; a persistence surface is an integrable function defined as a weighted sum of Gaussian functions, one centered at each point in the persistence diagram. If a subdomain of $\rho_B$ is passed under discretization, the result defines a grid. A persistence image can be created by computing the integral of $\rho_B$ on each grid box [3].
For the creation of persistence images, 3 values are chosen:

- the resolution, corresponding to the grid being overlaid on the persistence diagram;
- the distribution, being the choice of a probability distribution to assign to each point of the persistence diagram;
- the weighting function, responsible of the persistence surface from the persistent diagram, ensuring its stability.

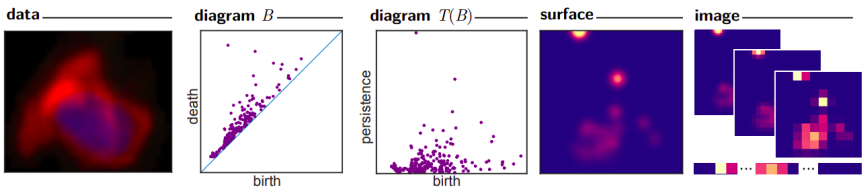



Figure 1.11: Pipeline from data to persistence image. Image from [3], original data and image worked on from [17]

One of the main advantages of persistence images is its ability to convey persistence diagrams of different homological dimensions into a single object [3].

## 1.7 Cubical persistence

As our data set is in the form of fMRIs, a persistence study directly on the voxels, modeling them with cubical complexes, may be more convenient. This approach has already been taken in similar cases [37]. The strength of this method is the reduction of the complex size, since triangulation of the the space is not necessary, and the possibility to use more compact data structures [40]. This approach tends to be efficient and scale better than standard persistent homology. Instead of triangulations, the space is worked

in cubical complexes. This type of complexes allow the usage of more compact data structures.
As the name suggests, cubical complexes are collections of cubes, the dimension of the complex is given by the highest dimension of its components and, as for simplicial complexes, it must be closed under taking faces and intersections.

**Definition 11.** *A **cube** is a product of d elementary intervals in a given d-space. The dimension of a cube is given by the number of unit intervals, $[k, k+1]$, of the cube. It follows that 0-cubes are vertices, 1-cubes are edges. 2-cubes are squares and 3-cubes are voxels.*
*A **face** of a cube is found only when we have two cubes $a, b \subseteq R^d$ and $a \subseteq b$; we say that a is a face of b.*

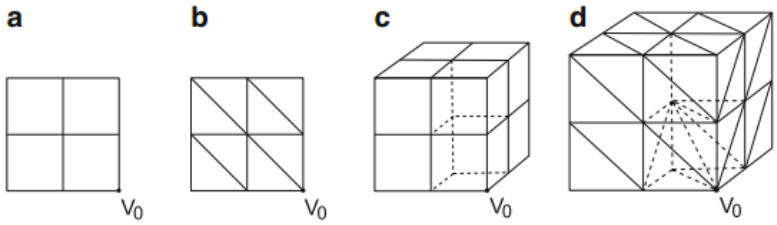



Figure 1.12: a) shows a 2D cubical complex, b) its triangulation, c) a 3D cubical complex, d) its triangulation. Image from [40]

## 1.8 A brief history of TDA application with Machine Learning in the medical field

The main focus of this thesis is to test the application of TDA techniques to the study of fMRI and neuroscience in general. This could lead to an innovation in the field of medicine and, possibly, help in the recognition of brain conditions.
Machine learning is becoming more and more present in the daily life resulting in numerous ethical discussions. As users we can decide to apply it consciously; it is a great tool if well handed, as for most thing the danger of it doesn't come from the tool itself but from its usage.
Following are some cases in the medical field that took benefice from the application of the topological analysis. TDA prospects itself as a good tool to analyze and classify blood clots which could help in the research about hemostasis and thrombosis [6]. TDA has found itself successful in the discovery of a subgroup of breast cancers [32]. The combination of appropriate techniques of both persistent homology and analysis of variance results in a better understanding of the data's nonlinear features in the study of

orthodontic data [26]. TDA has also been effective in the detection of genes with periodic profile [13]. A similar study to the one done here has been made in the university of Edinburgh and can recognize Alzheimer patients via the usage of Betti Curves [38].

## 1.9 Goal of application of TDA in our study

This thesis is a work structured over two main levels: firstly it handles the analysis of one preexisting pipeline for the treatment of fMRI data; it then proposes an adaptation of this pipeline studying the effects of PCA (Principal Component Analysis) coexisting with TDA analysis.
TDA is employed in this study to capture the underlying topological structure of high-dimensional fMRI data. By transforming neurological signals into persistent diagrams, TDA provides a compact representation of the data that aims to preserve relevant structural information. These topological descriptors are subsequently used as input features for a binary classification task, and the performance of the model is evaluated using standard metrics.
The goal of this analysis is to assess whether the inclusion or removal of PCA affects the ability of TDA-based features to support the classification task, thereby offering insight into the impact of dimensionality reduction on topological representations of fMRI data. In this study, the data represent two populations: subjects affected by neurodegenerative disorders and healthy controls. While the present work is limited to this specific setting, the proposed approach may be extended to other applications in neurological data analysis.

# Chapter 2

# fMRI data and basics neuroscience concepts

## 2.1 Introduction of the data

What we are working with are functional magnetic resonance imaging (fMRI). In this technology, the focus is on the study of blood oxygen measurements of 3D brain volumes divided into voxels. Thanks to this division, the fMRI image is already clustered. [4]
The dataset on which we work is the ENIGMA[1] (Enhancing Neuro Imaging Genetics through Meta Analysis) one.

## 2.2 Connectomic features and Regions of interest

### 2.2.1 Functional connectivity vectors

Functional connectivity has been defined as the correlations between spatially remote neurophysiological events ([31]). The functional relationships between different parts of the brain are examined and analyzed ([18]). This allows us to have an overview of the region of the brains that are activated in certain scenarios and how they relate with other areas. The intensity of this connections can also describe some more developed areas. The focus is no longer in singular regions, but on the ones that cooperate [20].
In the context of functional connectivity networks, connected components ($\beta_0$) may reflect the fragmentation or integration of brain subnetworks, while 1-dimensional holes ($\beta_1$) can be interpreted as recurrent connectivity loops or redundant communication pathways. Alterations in these structures may indicate disrupted information flow, which is commonly

[1] https://enigma.ini.usc.edu/

reported in neurodegenerative disorders. [34] To study these features we analyze regions of interest (ROI) over time.

### 2.2.2 Regions of interest

The data studied are already divided in regions of interest. We keep this division in our study to benefice of conducting a ROI analysis; in the particular, it gives some structure to the data and help identify and map the patterns of activities [36]. In our study, the ROIs are based on the Schaefer parcellation, having more than 400 ROIs residing in 17 resting state networks.

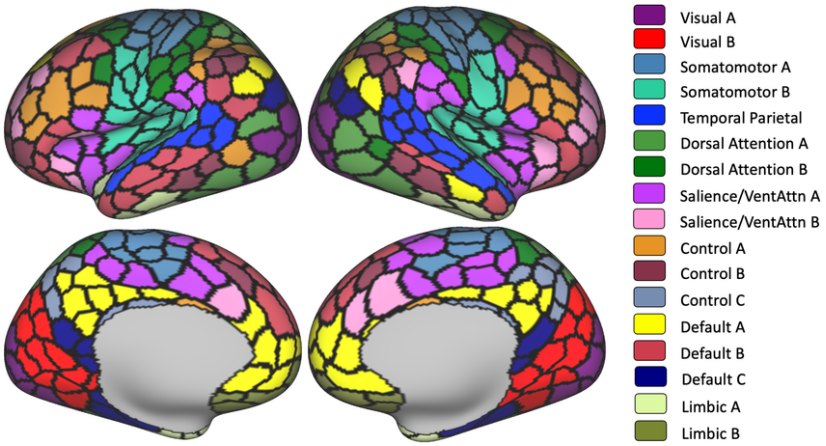



Figure 2.1: Schaefer parcellation of the brain. Image from [7]

### 2.2.3 Amplitude of Low-Frequency Fluctuations(ALFF) and Fractional Amplitude of Low-Frequency Fluctuations(fALFF)

Synchronous low frequency fluctuation (LFF) in resting-state functional magnetic resonance imaging (fMRI) has been used extensively to study functional brain activity. [42]

Amplitude of low-frequency fluctuations (ALFF) and fractional ALFF (fALFF) are resting-state functional MRI (rs-fMRI) metrics that quantify the power of spontaneous, low-frequency ( 0.01–0.10 Hz) fluctuations of the BOLD signal within a voxel or region of interest. ALFF measures the square root of the power spectrum within a predefined low-frequency band (commonly 0.01–0.08 or 0.01–0.10 Hz). The exact upper cutoff depends on the sampling rate (repetition time, TR) and must be below the Nyquist frequency $(1/(2 \cdot \mathrm{TR}))$ [44]

# Part II

# Implementation

# Chapter 3

# Previous work on the data

The data used in the thesis have already been studied previously. This work, done by Hanyang a master student in Munich, is used as a base to approach the dataset.

## 3.1 Previous TDA

In the specific two already existing python programs are studied: *FeatPrep.py* and *XGBoost.py*. They define the base pipelines' structure of the whole project.

### 3.1.1 FeatPrep.py

Feature preparation pipeline for the data to be treatable, meaning that it computes the preparation of homological components to then work statistically and in machine learning terms. Its main functions do the following:

- **fALFF** from a given time series, returns its fALFF and ALFF
- **betti_curve** as the name suggests, it returns the betti curves, so the Betti number at each t-value
- **persistence_landscape** returns the landscape of persistent diagram data
- **persistence_silhouette** returns the silhouettes of persistent diagram data
- **persistence_image** returns the persistence images from array of persistence intervals

The persistence diagrams at the base of the creation of these visualizations are of the form [birth, death, dim], where birth and death represent the persistency pairs and dim represent the homology dimension.

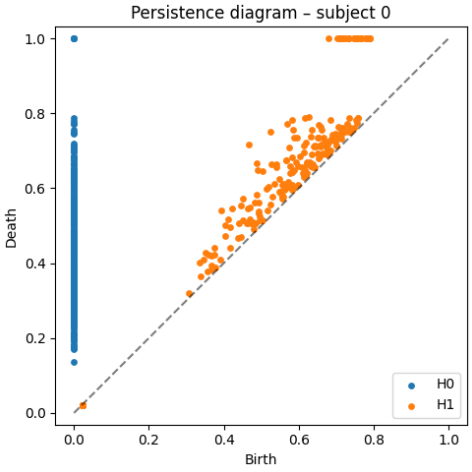
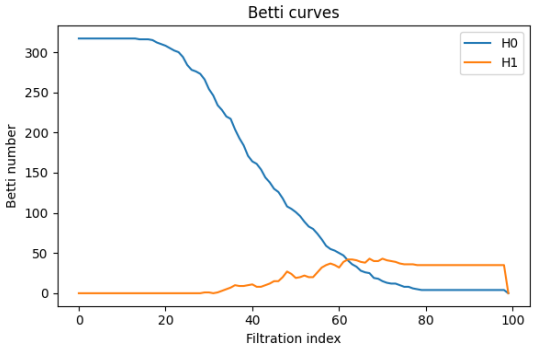
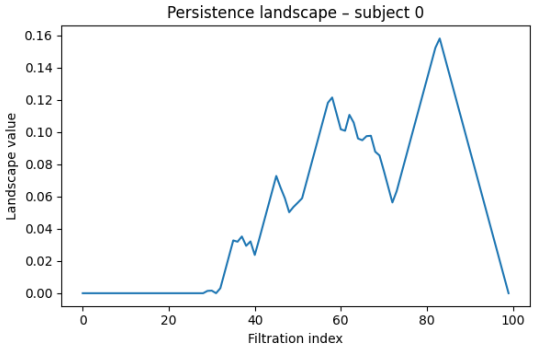


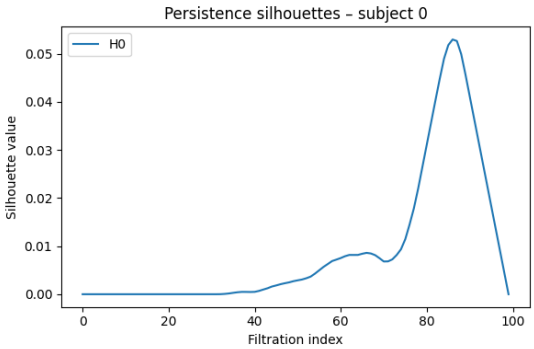

Figure 3.1: Persistence diagram from the given data for a studied subject



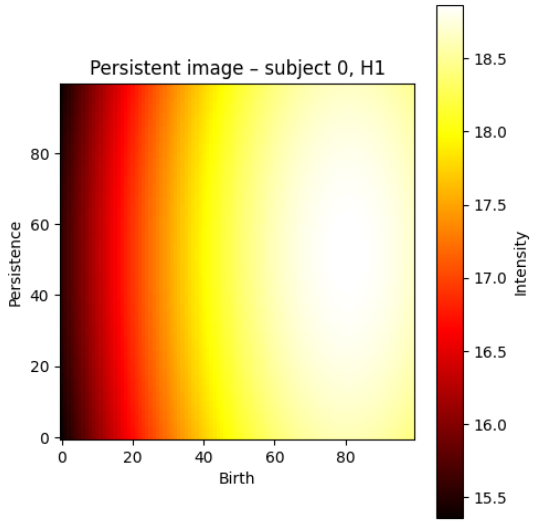

(a) Betti Curves




(b) Persistence landscape




(c) CPersistence silhouettes




(d) Persistent image of homology groups H1

Figure 3.2: Different visualization tools made applying different techniques to the persistence diagram of subject 0

In addition, the program also does many computations well useful for the further usage in machine learning models. These include the conversion of functional connectivity matrices into distance matrices suitable for Vietoris–Rips complexes; thresholding of FC matrices to remove weak connections, which reduces noise in TDA computations. Support for both ROI-level and network-level aggregation of features. Normalization and filtering of persistence diagrams to remove intervals with very short lifetimes ($death - birth < 0.01$), ensuring that the extracted topological descriptors capture robust features. Generation of feature arrays.
Some key parameters and their values are:

Table 3.1: Key parameters used in *FeatPrep.py*

| **Parameter** | **Value** | **Description** |
|---|---|---|
| TR | Subject-specific | Repetition time of fMRI acquisition, used in fALFF calculation |
| Low-frequency band | (0.01, 0.1 Hz) | Frequency range for fALFF computation |
| Fractional ALFF | True | Returns normalized fALFF if True; else raw ALFF |
| FC threshold percentile | 90 | Minimum percentile to retain connections in functional connectivity matrices |
| Minimum persistence | 0.01 | Filters out persistence intervals where $death - birth < 0.01$ |
| Homology dimensions | [0,1] | Compute H0 (connected components) and H1 (loops) in the Vietoris-Rips complex |
| Number of samples | 100 | Number of points to sample along the filtration axis for Betti curves, landscapes, silhouettes |
| Value p | 1 | value of trade-off parameter between uniformly treating all pairs in the persistence diagram and considering only the most persistent pairs in power-weighted silhouettes |
| Persistence image resolution | (100,100) | Grid size for converting persistence diagrams into persistence images |
| Persistence image sigma | 1.0 | Standard deviation of the Gaussian kernel applied in persistence images |
| Persistence image weight power | 1 | Exponent for weighting persistence intervals when constructing persistence images |

## 3.2 XGBooster

The training of the prepared data is done with the help of FLAML library[1] and using the XGBoost (eXtreme Gradient Boosting)[2] model.

[1] https://microsoft.github.io/FLAML
[2] https://xgboost.readthedocs.io/en/stable/

The data classified are the ones generated in *FeatPrep.py*, the input given to the model is then: the functional connectivity, the TDA descriptors (persistence landscapes, persistence silhouettes, Betti curves and persistence images) and the concatenations of functional connectivity with TDA descriptors. Seven subjects are excluded from the analysis following preliminary data quality assessment.
The *XGBooster.py* pipeline defines some functions able to perform the following actions:

- subject-level **z-scoring** (or normalization) across features, applied independently to each subject;
- **repeated stratified cross-validation** with 5-folds and 3 repetitions. Stratification ensures the stability of preexisting groupings, performing on the center where data were aquired and the diagnostic of the patient as defined in the table *DemTable*;
- **PCA dimensionality reduction** for any feature sets having more features than the proposed threshold (100 features in FLAML and more than 30 in XGBooster)

They are later called inside the two different modeling approaches *RepeatedStratifiedAutoML* and *RepeatedStratifiedXGB*. They are both based on XGBoost with explicit hyperparameter optimization.
For each feature configuration, the models return the mean accuracy and mean ROC_AUC (Area Under the Receiver Operating Characteristic Curve) averaged across all folds and repetitions.

### 3.2.1 PCA: Principal Component Analysis

PCA is a dimensionality reduction technique, its usage reduces the number of features in a data set maintaining the most important information. During this process data structures are transformed and rearranged. This is done by projecting high-dimensional data into a lower-dimensional linear subspace [27]
In the *XGBooster.py* pipeline, PCA reduces sets with more than a threshold components to the threshold value. That is 100 for FLAML-based models and 30 for XGBoost-based models. This step is intended to mitigate the effects of high dimensionality, such as noise amplification and overfitting, and to improve computational efficiency.
Since the features used in this study are derived from Topological Data Analysis (TDA), the application of PCA represents an interesting design choice. While PCA preserves most of the information of higher dimensions, it does not explicitly account for the topological structure encoded in TDA descriptors. For this reason, this thesis investigates the impact of including or excluding PCA on the performance of TDA-based classification models. account

### 3.2.2 FLAML

After the data have followed the procedure of cross-validation, normalization and PCA reduction, with the FLAML ability and some minimal manual settings, an optimized Hyperparameter is found to work on the data.

```
#FLAML settings
    automl = AutoML()
    automl_settings = {
        "time_budget": 60,  # seconds; adjust as needed
        "metric": "accuracy",
         "task": "classification",
        "estimator_list": [estimator],
        "log_file_name": "flaml.log",
        "verbose": 1,
        "n_jobs": -1,  # use all available cores
    }
```

This allows the computation of accuracy values and ROC_AUC values.

### 3.2.3 XGB

Similarly to the FLAML model, the data follow the procedure of cross-validation, normalization and PCA reduction; afterwards the hyperparameter to perform the last computations is found using the XGBooster classifier. When running the program, the user is able to decide whether to work with an XGBooster classifier using default hyperparameter or tuned one (choice made by the corresponding boolean value *search*).

```
#XGBoost with different procedures
    xgb = XGBClassifier(objective='binary:logistic')
    if search is True:
        params = {
            'min_child_weight': [1, 5, 10],
            'gamma': [0.5, 1, 1.5, 2, 5],
            'subsample': [0.6, 0.8, 1.0],
            'colsample_bytree': [0.6, 0.8, 1.0],
            'max_depth': [3, 4, 5]
            }
        model = RandomizedSearchCV(xgb, param_distributions=params,
   n_iter=5, scoring='roc_auc', n_jobs=4, cv=skf.split(X_train_z,
   grouplabels_train), verbose=2)
    else:
        model = xgb
```

### 3.2.4 Output

The output returned allows for a general overview of the XGBoost models; accuracies and ROC_AUC values stay below 0.60 for the tested configurations.
For all the models tried the values of accuracy and ROC_AUC are $\sim 0.5$ for most of the folds, hence it is not recognizing features from the data. The models have similar tendencies not showing an overall better performance distinguishing one of them. The ROC_AUC values can be better by a tiny amount; not enough to be kept into account. The output are relative to the features derived in *FeatPrep.py*. The pipeline outputs accuracy and ROC_AUC for models trained on the following feature sets:

- **Functional connectivity features**: network-level functional connectivity vectors (FCVectors_Net; 1 iteration), whole-brain functional connectivity vectors (FCVectors; 1 iteration), fALFF (1 iteration), and network-level fALFF (fALFF_Net; 1 iteration).
- **Topological data analysis (TDA) descriptors** evaluated per channel:Persistence images (PI; 2 iterations corresponding to homology dimensions 0 and 1); Persistence landscapes (PL; 100 iterations corresponding to landscape levels 0–99); Persistence silhouette (PS; 1 iteration); Betti curves (BC; 2 iterations corresponding to homology dimensions 0 and 1).
- **Concatenated TDA descriptor**s: PL_Concat (1 iteration), PI_Concat (1 iteration), BC_Concat (1 iteration), and PS_Concat (1 iteration).
- **Functional connectivity–augmented TDA descriptors concatenations**: PL_FCConcat (1 iteration), PI_FCConcat (1 iteration), BC_FCConcat (1 iteration), and PS_FCConcat (1 iteration).

The results reported here are rounded at $10^{-3}$. For all the levels of persistent landscapes, the results did not vary that is the reason why only one value per model is presented in the tables. The models seem to perform the worst with persistence landscapes, FLAML performs slightly better than XGBoost on the majority of results but on the application over Betti curves concatenation, persistence images concatenation and persistence silouhettes concatenation with functional connectivity features.

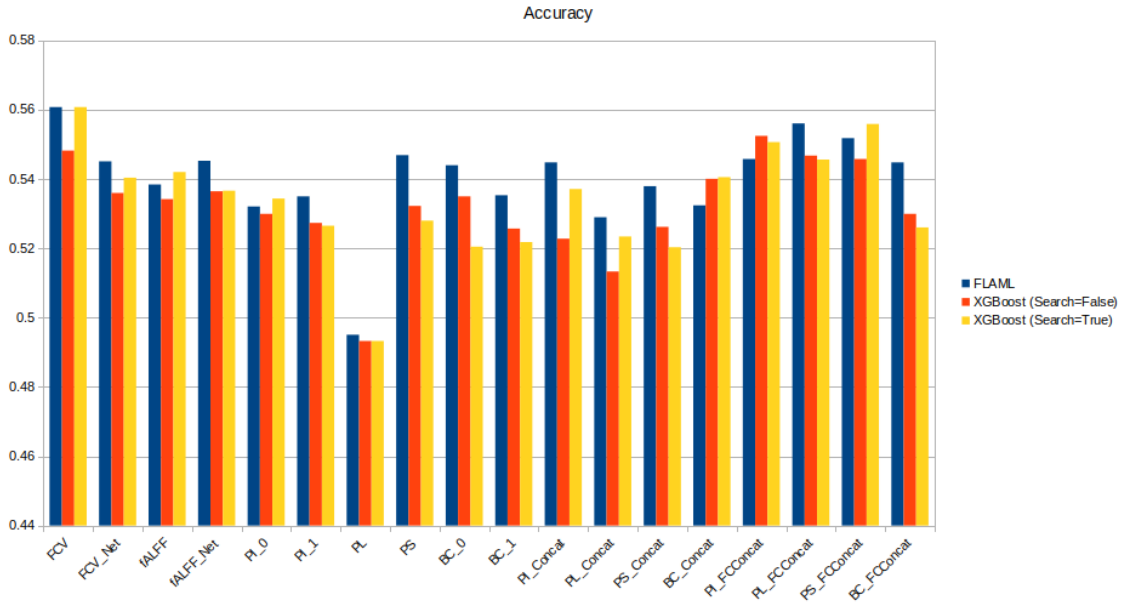



Figure 3.3: Visualization of mean of accuracy per region

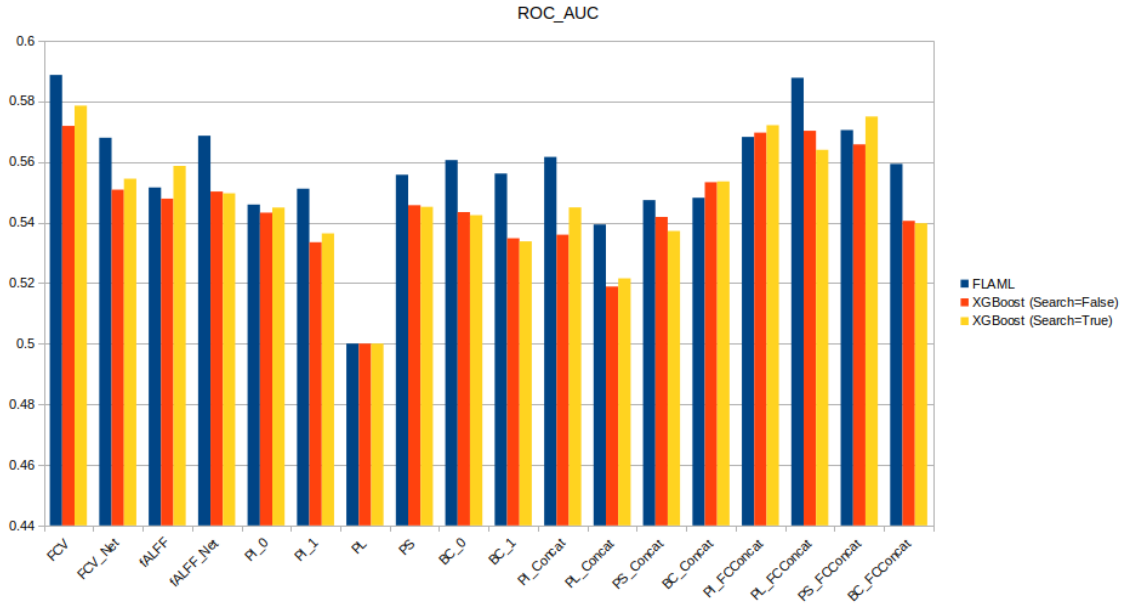



Figure 3.4: Visualization of mean of ROC_AUC per region

Table 3.2: Original pipeline: accuracy results, mean and standard deviation values

| | **FLAML** | | **XGBoost (Search = False)** | | **XGBoost (Search = True)** | |
|---|---|---|---|---|---|---|
| **Features** | **Mean** | **Std** | **Mean** | **Std** | **Mean** | **Std** |
| FCV | 0.561 | 0.030 | 0.548 | 0.021 | 0.561 | 0.019 |
| FCV_Net | 0.545 | 0.017 | 0.536 | 0.019 | 0.540 | 0.023 |
| fALFF | 0.538 | 0.024 | 0.534 | 0.023 | 0.542 | 0.027 |
| fALFF_Net | 0.545 | 0.022 | 0.536 | 0.024 | 0.537 | 0.025 |
| PI_0 | 0.532 | 0.023 | 0.530 | 0.026 | 0.534 | 0.024 |
| PI_1 | 0.535 | 0.027 | 0.527 | 0.025 | 0.526 | 0.028 |
| PL | 0.495 | 0.008 | 0.493 | 0.010 | 0.493 | 0.007 |
| PS | 0.547 | 0.022 | 0.532 | 0.021 | 0.528 | 0.024 |
| BC_0 | 0.544 | 0.019 | 0.535 | 0.028 | 0.520 | 0.021 |
| BC_1 | 0.535 | 0.024 | 0.526 | 0.020 | 0.522 | 0.029 |
| PI_Concat | 0.545 | 0.026 | 0.523 | 0.022 | 0.537 | 0.027 |
| PL_Concat | 0.529 | 0.025 | 0.513 | 0.023 | 0.523 | 0.024 |
| PS_Concat | 0.538 | 0.028 | 0.526 | 0.027 | 0.520 | 0.025 |
| BC_Concat | 0.532 | 0.021 | 0.540 | 0.023 | 0.541 | 0.025 |
| PI_FCConcat | 0.546 | 0.022 | 0.552 | 0.020 | 0.551 | 0.027 |
| PL_FCConcat | 0.556 | 0.024 | 0.547 | 0.023 | 0.546 | 0.023 |
| PS_FCConcat | 0.552 | 0.021 | 0.546 | 0.022 | 0.556 | 0.025 |
| BC_FCConcat | 0.545 | 0.023 | 0.530 | 0.025 | 0.526 | 0.024 |

Table 3.3: Original pipeline: ROC_AUC results, mean and standard deviation values

| | **FLAML** | | **XGBoost (Search = False)** | | **XGBoost (Search = True)** | |
|---|---|---|---|---|---|---|
| **Features** | **Mean** | **Std** | **Mean** | **Std** | **Mean** | **Std** |
| FCV | 0.589 | 0.030 | 0.572 | 0.033 | 0.579 | 0.022 |
| FCV_Net | 0.568 | 0.021 | 0.551 | 0.021 | 0.554 | 0.025 |
| fALFF | 0.552 | 0.022 | 0.548 | 0.024 | 0.559 | 0.026 |
| fALFF_Net | 0.569 | 0.024 | 0.550 | 0.023 | 0.550 | 0.027 |
| PI_0 | 0.546 | 0.029 | 0.543 | 0.031 | 0.545 | 0.032 |
| PI_1 | 0.551 | 0.026 | 0.533 | 0.024 | 0.536 | 0.027 |
| PL | 0.500 | 0.001 | 0.500 | 0.000 | 0.500 | 0.000 |
| PS | 0.556 | 0.025 | 0.546 | 0.026 | 0.545 | 0.021 |
| BC_0 | 0.561 | 0.018 | 0.543 | 0.028 | 0.542 | 0.021 |
| BC_1 | 0.556 | 0.025 | 0.535 | 0.026 | 0.534 | 0.032 |
| PI_Concat | 0.562 | 0.026 | 0.536 | 0.026 | 0.545 | 0.033 |
| PL_Concat | 0.539 | 0.025 | 0.519 | 0.028 | 0.522 | 0.024 |
| PS_Concat | 0.547 | 0.031 | 0.542 | 0.025 | 0.537 | 0.028 |
| BC_Concat | 0.548 | 0.024 | 0.553 | 0.022 | 0.554 | 0.026 |
| PI_FCConcat | 0.568 | 0.024 | 0.570 | 0.024 | 0.572 | 0.026 |
| PL_FCConcat | 0.588 | 0.025 | 0.570 | 0.023 | 0.564 | 0.028 |
| PS_FCConcat | 0.570 | 0.030 | 0.566 | 0.020 | 0.575 | 0.033 |
| BC_FCConcat | 0.559 | 0.023 | 0.541 | 0.026 | 0.540 | 0.036 |

# Chapter 4

# Adaptation of preexisting work

Starting from the Python pickle file containing the data, the same pipeline for the preparation of the data via TDA methods is kept, but a different machine learning pipeline is tested. The goal is to try and improve the existing one.

## 4.1 Change in XGBoost pipeline, removal of PCA

Given the properties of Principal Component Analysis (PCA), an alternative XGBoost pipeline without dimensionality reduction is created and investigated. The main motivation behind this choice is to evaluate whether PCA excessively alters the feature space derived from Topological Data Analysis (TDA), potentially discarding crucial information, and to quantify the impact of PCA on model performance.
The same topological descriptors employed in the original *XGBooster.py* pipeline are retained. The only modification concerns the machine learning pipeline: normalized feature folds are no longer reduced through PCA. Consequently, both the AutoML-based and XGBoost-based classifiers operate directly on high-dimensional TDA-derived feature representations. The modifications are represented in 4.1.

### 4.1.1 Expected outcomes

Removing PCA may preserve subtle topological patterns that are otherwise mixed or attenuated by projection into lower-dimensional space. The interest of this new analysis is then to investigate whether the space reduction has a remarked consequence on the TDA-features classified by the models.
Topological analysis tends to be a good solution for non-linear data [19], making it theoretically appropriate for the preparation of data in our set up. What is examined, then, is whether the application of PCA may have had an impact on the performance of the

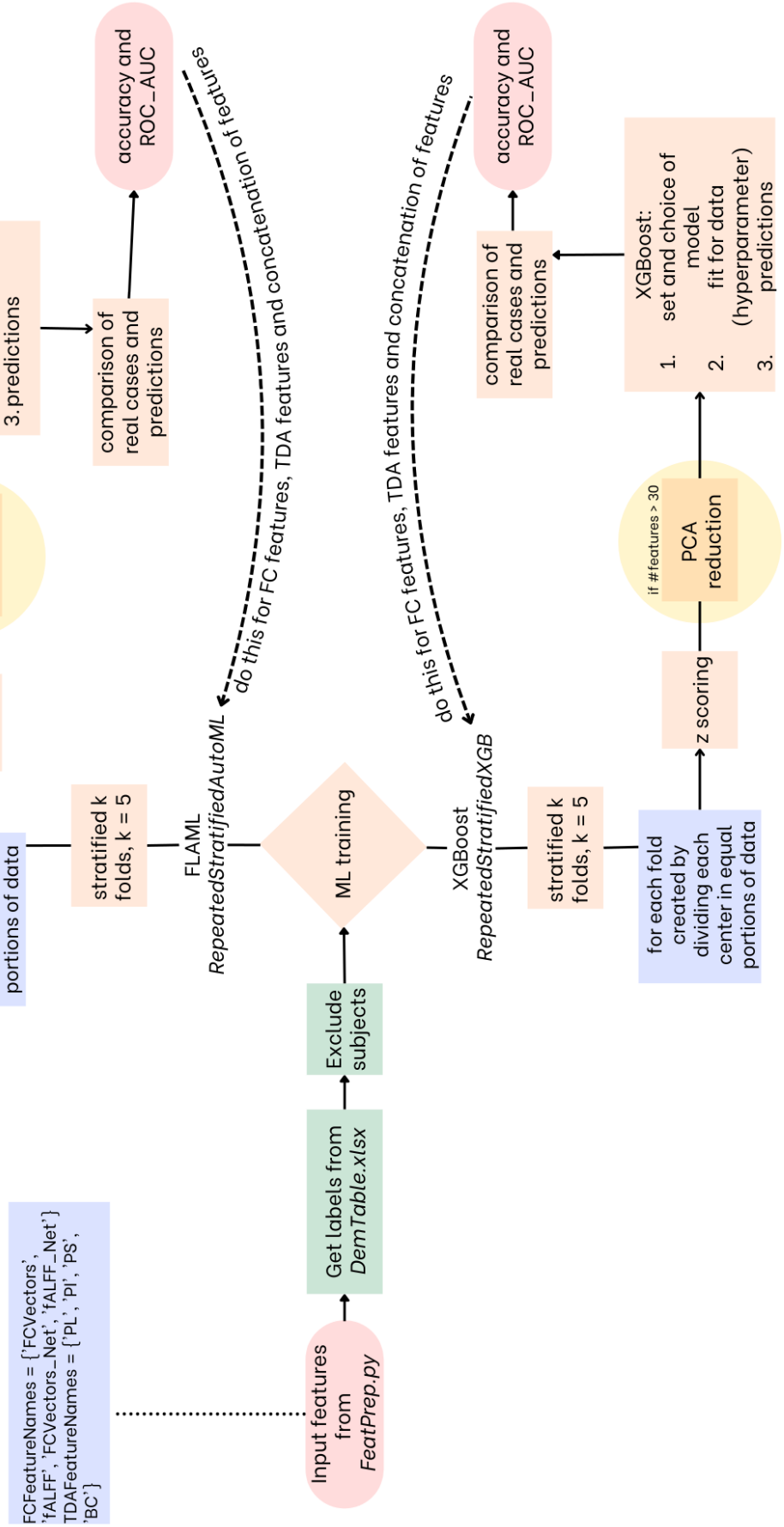



Figure 4.1: Pipeline of XGBoost, in the proposed modification the PCA reduction steps marked by yellow circles are not included

models making the learned representations less adapted to the underlying data.

### 4.1.2 Effects of higher dimensionality analysis

Working with high-dimensional data introduces several mathematical and computational challenges, such as the difficulty of accurately approximating or integrating high-dimensional functions [14]. However, high-dimensional spaces also exhibit specific statistical phenomena, including concentration of measure, whereby certain random fluctuations become more predictable and structured than in lower-dimensional settings.
In the context of this study, operating directly in the high-dimensional space of TDA-derived features may allow subtle but consistent topological patterns to be preserved and exploited by the classification models, rather than being compressed or distorted through dimensionality reduction.

### 4.1.3 Outcomes of the new pipeline

After running the new pipeline, the same graphs as for the original one are plotted. They can be find in *Annexes*. The results are quite consistent among the models both in accuracy and ROC_AUC results. ROC_AUC can reach slightly higher values than accuracy, but never reach the threshold of 0.6. As for the original pipeline, the standard deviation is around 0.02.

Table 4.1: New pipeline: accuracy results, mean and standard deviation

| | **FLAML** | | **XGBoost (Search = F)** | | **XGBoost (Search = T)** | |
|---|---|---|---|---|---|---|
| **Features** | **Mean** | **Std** | **Mean** | **Std** | **Mean** | **Std** |
| FCV | 0.550 | 0.029 | 0.549 | 0.030 | 0.558 | 0.019 |
| FCV_Net | 0.556 | 0.018 | 0.539 | 0.016 | 0.533 | 0.021 |
| fALFF | 0.544 | 0.022 | 0.536 | 0.022 | 0.540 | 0.025 |
| fALFF_Net | 0.547 | 0.023 | 0.548 | 0.023 | 0.546 | 0.025 |
| PI_0 | 0.551 | 0.024 | 0.534 | 0.022 | 0.535 | 0.025 |
| PI_1 | 0.528 | 0.026 | 0.535 | 0.024 | 0.532 | 0.027 |
| PL | 0.495 | 0.007 | 0.493 | 0.010 | 0.493 | 0.008 |
| PS | 0.540 | 0.021 | 0.535 | 0.022 | 0.524 | 0.025 |
| BC_0 | 0.545 | 0.020 | 0.535 | 0.029 | 0.527 | 0.022 |
| BC_1 | 0.537 | 0.023 | 0.526 | 0.021 | 0.528 | 0.029 |
| PI_Concat | 0.546 | 0.025 | 0.523 | 0.024 | 0.524 | 0.027 |
| PL_Concat | 0.529 | 0.024 | 0.512 | 0.023 | 0.510 | 0.024 |
| PS_Concat | 0.541 | 0.029 | 0.535 | 0.027 | 0.533 | 0.024 |
| BC_Concat | 0.558 | 0.021 | 0.541 | 0.022 | 0.542 | 0.025 |
| PI_FCConcat | 0.559 | 0.022 | 0.549 | 0.021 | 0.550 | 0.027 |
| PL_FCConcat | 0.550 | 0.023 | 0.555 | 0.024 | 0.546 | 0.024 |
| PS_FCConcat | 0.553 | 0.021 | 0.555 | 0.020 | 0.551 | 0.024 |
| BC_FCConcat | 0.551 | 0.024 | 0.539 | 0.024 | 0.532 | 0.025 |

Table 4.2: New pipeline: ROC_AUC results, mean and standard deviation

| | **FLAML** | | **XGBoost (Search = F)** | | **XGBoost (Search = T)** | |
|---|---|---|---|---|---|---|
| **Features** | **Mean** | **Std** | **Mean** | **Std** | **Mean** | **Std** |
| FCV | 0.571 | 0.030 | 0.575 | 0.033 | 0.578 | 0.022 |
| FCV_Net | 0.578 | 0.021 | 0.555 | 0.021 | 0.550 | 0.025 |
| fALFF | 0.565 | 0.022 | 0.549 | 0.21 | 0.553 | 0.026 |
| fALFF_Net | 0.571 | 0.024 | 0.562 | 0.18 | 0.556 | 0.027 |
| PI_0 | 0.574 | 0.029 | 0.548 | 0.031 | 0.548 | 0.032 |
| PI_1 | 0.547 | 0.026 | 0.544 | 0.022 | 0.540 | 0.027 |
| PL | 0.500 | 0.000 | 0.500 | 0.000 | 0.500 | 0.000 |
| PS | 0.552 | 0.025 | 0.548 | 0.026 | 0.538 | 0.021 |
| BC_0 | 0.561 | 0.018 | 0.544 | 0.028 | 0.541 | 0.021 |
| BC_1 | 0.556 | 0.025 | 0.533 | 0.026 | 0.539 | 0.032 |
| PI_Concat | 0.567 | 0.026 | 0.534 | 0.026 | 0.533 | 0.033 |
| PL_Concat | 0.539 | 0.025 | 0.521 | 0.028 | 0.518 | 0.024 |
| PS_Concat | 0.549 | 0.031 | 0.544 | 0.025 | 0.550 | 0.028 |
| BC_Concat | 0.577 | 0.024 | 0.556 | 0.022 | 0.562 | 0.026 |
| PI_FCConcat | 0.570 | 0.024 | 0.571 | 0.024 | 0.569 | 0.026 |
| PL_FCConcat | 0.580 | 0.025 | 0.576 | 0.023 | 0.566 | 0.028 |
| PS_FCConcat | 0.575 | 0.030 | 0.571 | 0.020 | 0.567 | 0.033 |
| BC_FCConcat | 0.576 | 0.023 | 0.558 | 0.026 | 0.543 | 0.036 |

# Part III

# Results

# Chapter 5

# Results and remarks

## 5.1 Different methods outcomes

Evaluating the differences between the two pipelines, the first noticeable thing is how both have results of values around 0.50. The differences found are still very close to the original values; for accuracy the differences range from -0.02559087204564 to 0.010431947840261; for ROC_AUC the differences range from -0.029117776447547 to 0.017948680968493. The absolute difference is then of less than 0.03. This output clarify that the usage of PCA in the previous work did not have a big impact on the results, hence was not the limiting factor.
Studying the features one by one some trends are noticeable. The tendencies of accuracy and ROC_AUC values are the same, when one pipeline reveals herself better than the other for accuracy, it also does for ROC_AUC .The persistence landscapes keep the same value both for the original and the modified pipeline. This suggests that the lack of discriminative power of these descriptors is probably determined already before the machine learning pipeline, rather than being a consequence of dimensionality reduction or model choice. Persistence landscapes provide a global summary of the topological structure of functional connectivity, whereas neurodegenerative disorders are often characterized by localized or network-specific alterations. As a result, such global descriptors may fail to capture disease-related differences relevant for classification.

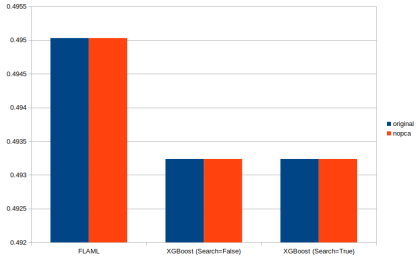



Figure 5.1: Accuracy values for PL

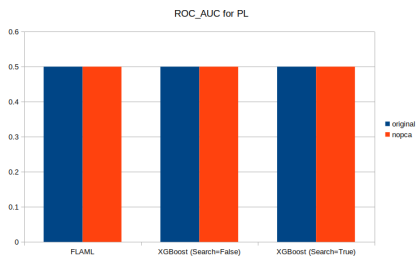



Figure 5.2: ROC_AUC values for PL

The effect on the functional connectivity features is not consistent between the models. The one model improving the values of accuracy and ROC_AUC for both functional connectivity and fALFF is XGBoost with default tuning of hyperparameter. The other two models do not show a clear tendency of preference for one pipeline or the other one. The improvements or declines are still of low difference from the original one, being of centesimal of percentage; this could be explained by just some noise reduction.

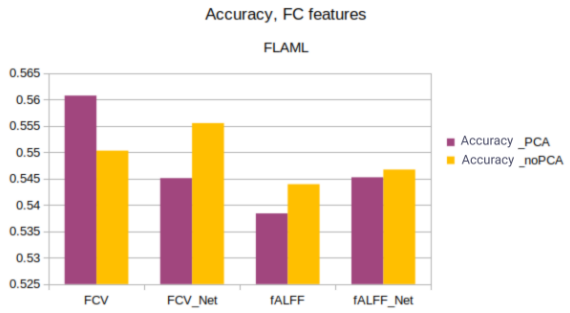



Figure 5.3: Accuracy values for FC features, FLAML model. Old pipeline on the left, new pipeline on the right.

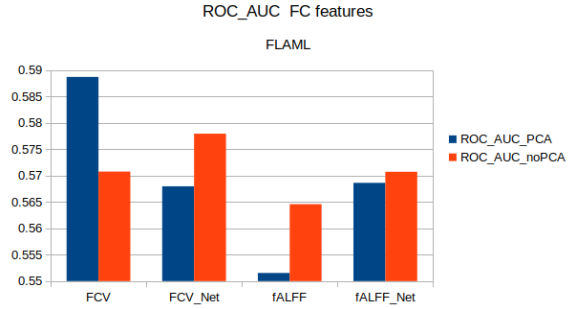



Figure 5.4: ROC_AUC values, FC features, FLAML model

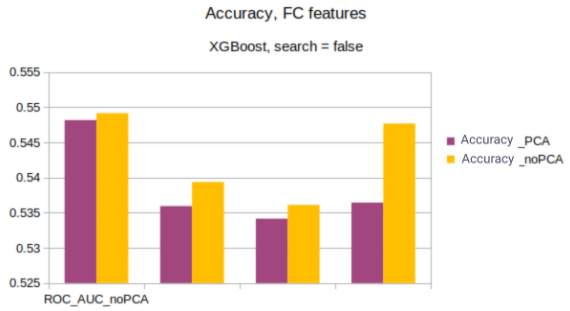



Figure 5.5: Accuracy values for FC features, XGBoost model. Old pipeline on the left, new pipeline on the right.

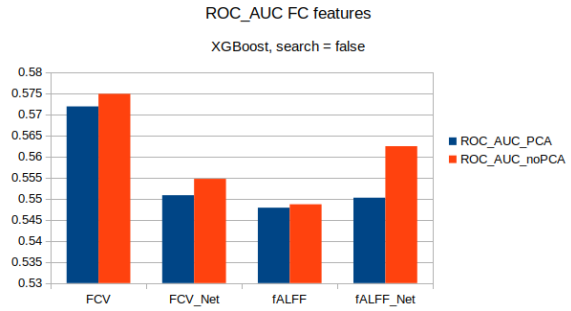



Figure 5.6: ROC_AUC values, FC features, XGBoost, search = false model

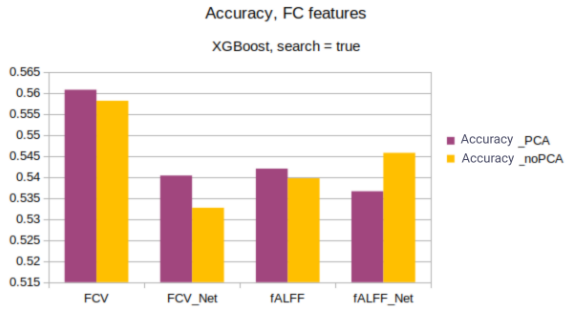



Figure 5.7: Accuracy values for FC features, XGBoost model with hyperparameter tuned on data. Old pipeline on the left, new pipeline on the right.

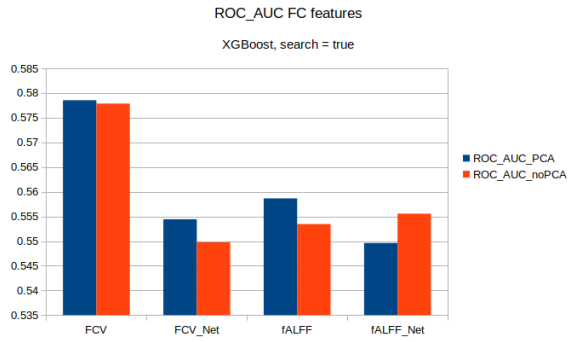



Figure 5.8: ROC_AUC values, FC features, XGBoost, search = true model

TDA features have different feedback accordingly to the descriptor. Persistent images

seems to be the one descriptor benefiting the most from the removal of PCA, while the other descriptors have inconsistent responses.

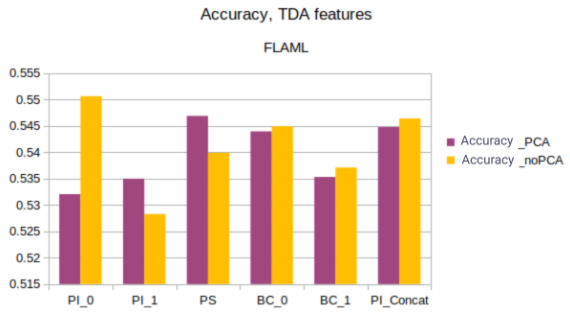



Figure 5.9: Accuracy values for TDA features, FLAML model. Old pipeline on the left, new pipeline on the right.

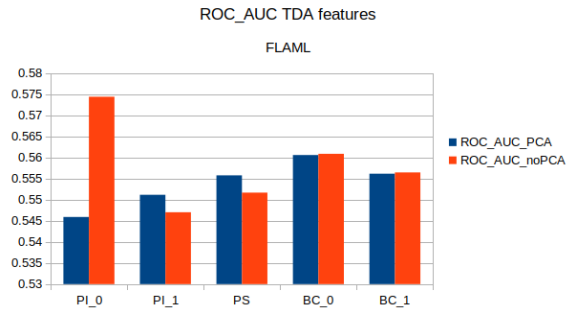



Figure 5.10: ROC_AUC values, TDA features, FLAML model

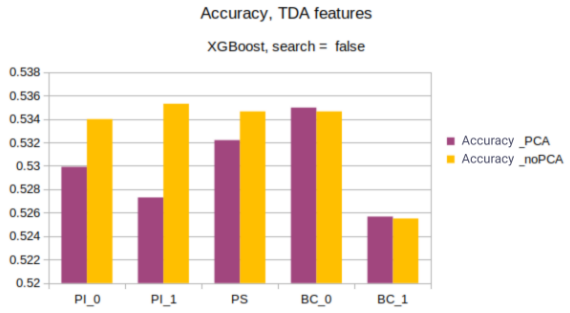



Figure 5.11: Accuracy values for TDA features, XGBoost model. Old pipeline on the left, new pipeline on the right.

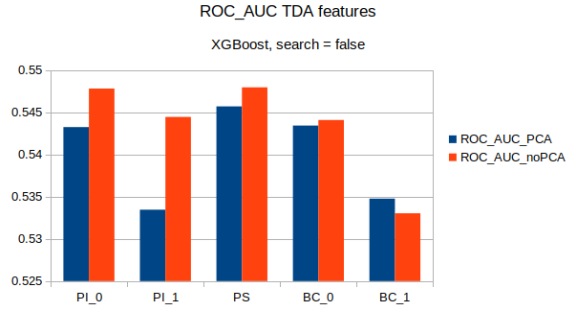



Figure 5.12: ROC_AUC values, TDA features, XGBoost, search = false model

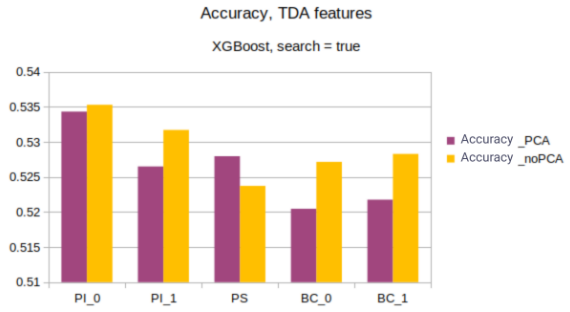



Figure 5.13: Accuracy values for TDA features, XGBoost model with hyperparameter tuned on data. Old pipeline on the left, new pipeline on the right.

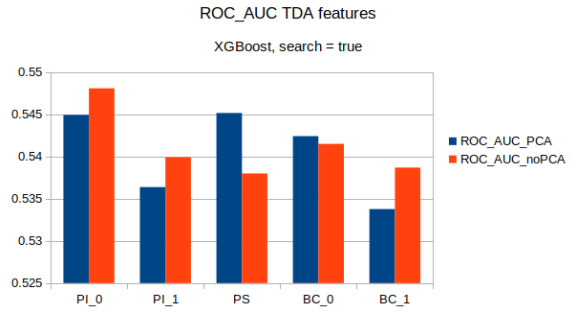



Figure 5.14: ROC_AUC values, TDA features, XGBoost, search = true model

Concatenations are very promising. Already in the original pipeline are between the feature sets with best performance results showing how TDA descriptors benefits from the combination with FC feature sets. The effect of removing PCA over the concatenation of TDA feature alone is not consistent. On the other hand we can observe how persistence silhouettes, as well as persistence landscapes, concatenations withing them-

selves and with FC features mostly improve the performance resulting in some of the highest values, with exceptions in XGBoost with search model. Betti curves have also a noticeable improvement when they are object of the non-PCA models. Persistence images have a little improvement in the first two models (FLAML and XGBoost without tune hyperparameter) but of little significance being of factor $< 10^{-2}$ of the percentage.

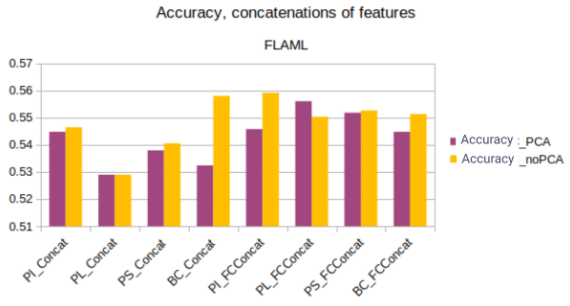



Figure 5.15: Accuracy values concatenations of TDA features with and without FC features, FLAML model. Old pipeline on the left, new pipeline on the right.

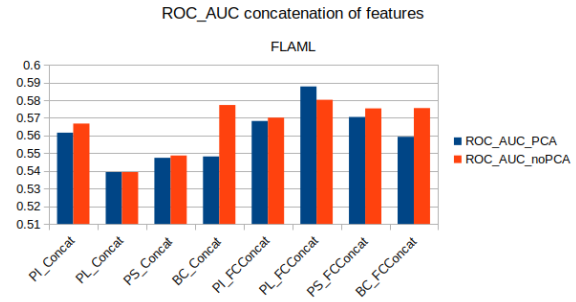



Figure 5.16: ROC_AUC values, concatenations of TDA features with and without FC features, FLAML model

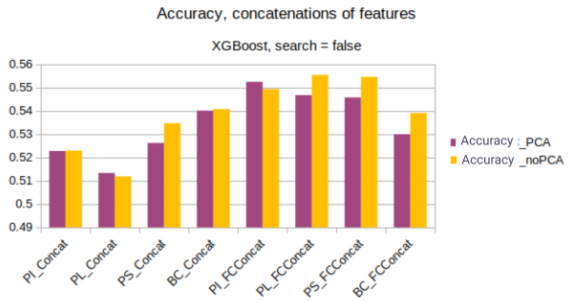



Figure 5.17: Accuracy values concatenations of TDA features with and without FC features, XGBoost model. Old pipeline on the left, new pipeline on the right.

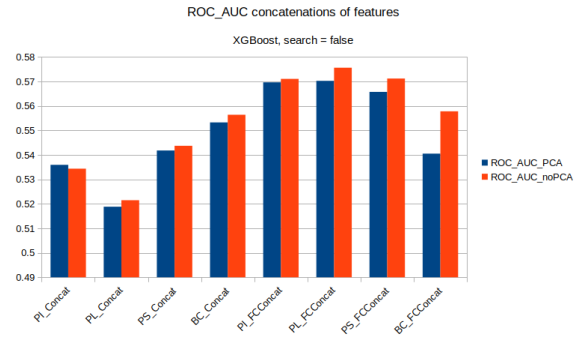



Figure 5.18: ROC_AUC values,concatenations of TDA features with and without FC features, XGBoost, search = false model

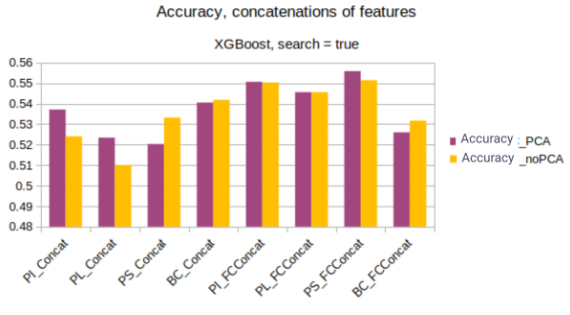



Figure 5.19: Accuracy values for concatenations of TDA features with and without FC features, XGBoost model with hyperparameter tuned on data. Old pipeline on the left, new pipeline on the right.

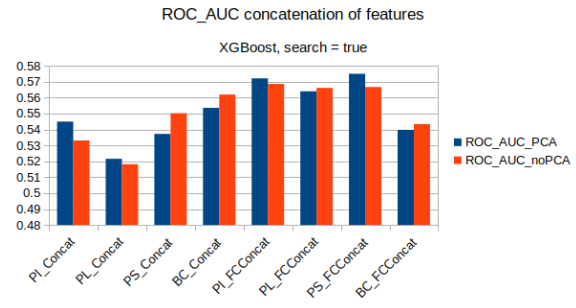



Figure 5.20: ROC_AUC values, concatenations of TDA features with and without FC features, XGBoost, search = true model

## 5.2 Possible explanation for the observed TDA performance

A key element of the obtained results is the strong consistency of the performance plateau around 0.5 across different classifiers, feature types, and experiments conducted with and without PCA. Such stability indicates that the observed lack of discriminative power is not attributable to the learning algorithms or to dimensionality reduction, but rather to the relationship between the extracted features and the diagnostic labels.
A plausible explanation is that the employed TDA descriptors primarily capture global geometrical and topological properties of functional brain networks, which are largely shared across individuals. Neurodegenerative diseases, however, are often characterized by localized or network-specific alterations that may not significantly affect the overall topological structure of functional connectivity. As a consequence, global topological summaries may fail to preserve disease-specific information required for binary classification.
Furthermore, although the combination of functional connectivity features with TDA descriptors represents an interesting hybrid approach, the results suggest that this integration does not sufficiently enhance class separability in the present setting. This may be due to the heterogeneity of neurodegenerative pathologies, which can obscure subtle disease-related effects in global representations.

## 5.3 Future improvements

The proposed method doesn't seem to be the solution to get better results and insights from the data. A reason behind this might be the fact that the change happens too late in the overall pipeline. Originally, PCA reduction is done after the TDA descriptors and functional connectivity features are already constructed and determined.
PCA doesn't result as the limiting factor. A way to get more different results would be working on the first preprocessing pipeline, rather than on the machine learning pipeline. This might cause more relevant changes of the features we use to train the model, and hence in the results.
Another remark made on data is how concatenations of features return slightly better values, so focusing in that direction could be beneficial.
As in the study of Alzheimer disease, the application of TDA separately to gray and white matter, when data allows it, as well as keeping into account the effect that the neurodegenerative diseases have on the brain and the comparison with age-related degeneration[38].

# Conclusion

This thesis explored the use of Topological Data Analysis as a feature extraction framework for resting-state fMRI data, with the goal of distinguishing neurodegenerative patients from healthy controls. Persistent homology was applied to functional connectivity matrices to derive a variety of topological descriptors, which were subsequently evaluated using gradient-boosting-based classification models.

Across all experimental configurations, including different classifiers, feature combinations, and the presence or absence of PCA-based dimensionality reduction, classification performance consistently remained non-deterministic. The main suggestion from this thesis is that future applications of TDA to neuroimaging data may benefit from focusing on earlier stages of the preprocessing pipeline, incorporating region-specific analyses, or explicitly accounting for disease heterogeneity and age-related effects. In this sense, the study contributes to a clearer understanding of the conditions under which TDA-based methods may or may not be effective in neuroimaging applications.

In conclusion, this thesis provides a systematic evaluation of TDA techniques applied to fMRI data and demonstrates the importance of critically assessing the assumptions underlying topological representations. While TDA alone may not suffice for binary neurodegenerative disease classification in the present setting, it remains a promising tool whose utility may emerge when combined with more targeted preprocessing strategies or complementary analytical approaches.

# Bibliography


[1] Michał Adamaszek and Henry Adams. The vietoris–rips complexes of a circle. *Pacific Journal of Mathematics*, 290(1):1–40, 2017.

[2] Michal Adamaszek, Henry Adams, Ellen Gasparovic, Maria Gommel, Emilie Purvine, Radmila Sazdanovic, Bei Wang, Yusu Wang, and Lori Ziegelmeier. Vietoris-rips and cech complexes of metric gluings. In *34th International Symposium on Computational Geometry (SoCG 2018)*, 2018.

[3] Henry Adams, Tegan Emerson, Michael Kirby, Rachel Neville, Chris Peterson, Patrick Shipman, Sofya Chepushtanova, Eric Hanson, Francis Motta, and Lori Ziegelmeier. Persistence images: A stable vector representation of persistent homology. *Journal of Machine Learning Research*, 18(8):1–35, 2017.

[4] Pulkit Agrawal, Dustin Stansbury, Jitendra Malik, and Jack L Gallant. Pixels to voxels: modeling visual representation in the human brain. *arXiv preprint arXiv:1407.5104*, 2014.

[5] Henry Adams Applied Algebraic Topology Network and Jan Segert; University of Florida. What is the difference between vietoris-rips and Čech complexes?, Aug 2020. YouTube video.

[6] Martin Berger, Tobias Hell, Anna Tobiasch, Judith Martini, Andrea Lindner, Helmuth Tauber, Mirjam Bachler, and Martin Hermann. Analysis of fibrin networks using topological data analysis–a feasibility study. *Scientific Reports*, 14(1):13123, 2024.

[7] John Bero, Yang Li, Aviral Kumar, Colin Humphries, Snehesh Nag, Heungyeol Lee, Woo Young Ahn, Sowon Hahn, Robert Todd Constable, Hackjin Kim, et al. Coordinated anatomical and functional variability in the human brain during adolescence. *Human Brain Mapping*, 44(4):1767–1778, 2023.

[8] Eric Berry, Yen-Chi Chen, Jessi Cisewski-Kehe, and Brittany Terese Fasy. Functional summaries of persistence diagrams. *Journal of Applied and Computational Topology*, 4(2):211–262, 2020.

[9] Peter Bubenik et al. Statistical topological data analysis using persistence landscapes. *J. Mach. Learn. Res.*, 16(1):77–102, 2015.

[10] Gunnar Carlsson. Persistent homology and applied homotopy theory. In *Handbook of homotopy theory*, pages 297–329. Chapman and Hall/CRC, 2020.

[11] Frédéric Chazal, Brittany Terese Fasy, Fabrizio Lecci, Alessandro Rinaldo, and Larry Wasserman. Stochastic convergence of persistence landscapes and silhouettes. In *Proceedings of the thirtieth annual symposium on Computational geometry*, pages 474–483, 2014.

[12] Yu-Min Chung and Austin Lawson. Persistence curves: A canonical framework for summarizing persistence diagrams. *Advances in Computational Mathematics*, 48(1):6, 2022.

[13] Mary-Lee Dequeant, Sebastian Ahnert, Herbert Edelsbrunner, Thomas MA Fink, Earl F Glynn, Gaye Hattem, Andrzej Kudlicki, Yuriy Mileyko, Jason Morton, Arcady R Mushegian, et al. Comparison of pattern detection methods in microarray time series of the segmentation clock. *PLoS One*, 3(8):e2856, 2008.

[14] David L Donoho et al. High-dimensional data analysis: The curses and blessings of dimensionality. *AMS math challenges lecture*, 1(2000).

[15] Herbert Edelsbrunner and John Harer. *Computational topology: an introduction*. American Mathematical Soc., 2010.

[16] Herbert Edelsbrunner, John Harer, et al. Persistent homology-a survey. *Contemporary mathematics*, 453(26):257–282, 2008.

[17] Tegan Emerson, Michael Kirby, Kelly Bethel, Anand Kolatkar, Madelyn Luttgen, Stephen O'Hara, Paul Newton, and Peter Kuhn. Fourier-ring descriptor to characterize rare circulating cells from images generated using immunofluorescence microscopy. *Computerized Medical Imaging and Graphics*, 40:70–87, 2015.

[18] Matthew Feigelis and Deanna J Greene. Functional connectivity in the gilles de la tourette syndrome. In *International Review of Movement Disorders*, volume 4, pages 103–125. Elsevier, 2022.

[19] Benjamin Alexander Fraser. *TDA parameters in time series analysis*. PhD thesis, Nipissing University, Faculty of Arts & Science, 2017.

[20] Karl J Friston, Chris D Frith, Peter F Liddle, and Richard SJ Frackowiak. Functional connectivity: the principal-component analysis of large (pet) data sets. *Journal of Cerebral Blood Flow & Metabolism*, 13(1):5–14, 1993.

[21] Ulderico Fugacci, Sara Scaramuccia, Federico Iuricich, Leila De Floriani, et al. Persistent homology: a step-by-step introduction for newcomers. In *STAG*, pages 1–10, 2016.

[22] Robert Ghrist. Barcodes: the persistent topology of data. *Bulletin of the American Mathematical Society*, 45(1):61–75, 2008.

[23] Rocío González-Díaz, María José Jiménez, Belén Medrano, and Pedro Real. A tool for integer homology computation: $\lambda$-at-model. *Image and Vision Computing*, 27(7):837–845, 2009.

[24] Kiya W Govek, Venkata S Yamajala, and Pablo G Camara. Clustering-independent analysis of genomic data using spectral simplicial theory. *PLoS computational biology*, 15(11):e1007509, 2019.

[25] Allen Hatcher. *Algebraic Topology*. Cambridge University Press, Cambridge, 2002.

[26] Giseon Heo, Jennifer Gamble, and Peter T Kim. Topological analysis of variance and the maxillary complex. *Journal of the American Statistical association*, 107(498):477–492, 2012.

[27] Ian Jolliffe. Principal component analysis. In *International encyclopedia of statistical science*, pages 1094–1096. Springer, 2011.

[28] CASEY KELLEHER and ALESSANDRA PANTANO. Introduction to simplicial complexes. *url: https://www. math. uci. edu/˜ mathcircle/materials/MCsimplex. pdf (cit. on p. 40)*.

[29] Richard NL Lamptey, Bivek Chaulagain, Riddhi Trivedi, Avinash Gothwal, Buddhadev Layek, and Jagdish Singh. A review of the common neurodegenerative disorders: current therapeutic approaches and the potential role of nanotherapeutics. *International journal of molecular sciences*, 23(3):1851, 2022.

[30] Jiayi Li and Cheng Zhao. Counting voids and filaments: Betti curves as a powerful probe for cosmology. *arXiv preprint arXiv:2512.07236*, 2025.

[31] Clas Linnman, Eric A. Moulton, Gabi Barmettler, Lino Becerra, and David Borsook. Neuroimaging of the periaqueductal gray: State of the field. *NeuroImage*, 60(1):505–522, 2012.

[32] Monica Nicolau, Arnold J Levine, and Gunnar Carlsson. Topology based data analysis identifies a subgroup of breast cancers with a unique mutational profile and excellent survival. *Proceedings of the National Academy of Sciences*, 108(17):7265–7270, 2011.

[33] Mathilde Papillon, Sophia Sanborn, Johan Mathe, Louisa Cornelis, Abby Bertics, Domas Buracas, Hansen J Lillemark, Christian Shewmake, Fatih Dinc, Xavier Pennec, et al. Beyond euclid: an illustrated guide to modern machine learning with geometric, topological, and algebraic structures. *Machine Learning: Science and Technology*, 6(3):031002, 2025.

[34] Alice Patania, Francesco Vaccarino, and Giovanni Petri. Topological analysis of data. *EPJ Data Science*, 6(1):1–6, 2017.

[35] Mariam Pirashvili, Lee Steinberg, Francisco Belchi Guillamon, Mahesan Niranjan, Jeremy G Frey, and Jacek Brodzki. Improved understanding of aqueous solubility modeling through topological data analysis. *Journal of cheminformatics*, 10(1):54, 2018.

[36] Russell A Poldrack. Region of interest analysis for fmri. *Social cognitive and affective neuroscience*, 2(1):67–70, 2007.

[37] 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.

[38] Ameer Saadat-Yazdi, Rayna Andreeva, and Rik Sarkar. Topological detection of alzheimer's disease using betti curves. In *International Workshop on Interpretability of Machine Intelligence in Medical Image Computing*, pages 119–128. Springer, 2021.

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

[40] Hubert Wagner, Chao Chen, and Erald Vuçini. Efficient computation of persistent homology for cubical data. In *Topological methods in data analysis and visualization II: theory, algorithms, and applications*, pages 91–106. Springer, 2011.

[41] Shmuel Weinberger. What is... persistent homology. *Notices of the AMS*, 58(1):36–39, 2011.

[42] Hong Yang, Xiang-Yu Long, Yihong Yang, Hao Yan, Chao-Zhe Zhu, Xiang-Ping Zhou, Yu-Feng Zang, and Qi-Yong Gong. Amplitude of low frequency fluctuation within visual areas revealed by resting-state functional mri. *Neuroimage*, 36(1):144–152, 2007.

[43] Afra Zomorodian. Fast construction of the vietoris-rips complex. *Computers & Graphics*, 34(3):263–271, 2010.

[44] Qi-Hong Zou, Chao-Zhe Zhu, Yihong Yang, Xi-Nian Zuo, Xiang-Yu Long, Qing-Jiu Cao, Yu-Feng Wang, and Yu-Feng Zang. An improved approach to detection of amplitude of low-frequency fluctuation (alff) for resting-state fmri: fractional alff. *Journal of neuroscience methods*, 172(1):137–141, 2008.

# Appendices

# .1 DemTable.xslx

| SubID | OrigID | Diag | Onset | Med | YBOCS | Age | AgeGroup | Sex | Edu | DepCurr | DepLife | AnxCurr | AnxLife | Agr_Check | Clean | Sex_Rel | Hoard | Ord | Center | TR |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AB_0_000 | sub-916005 | 2 | | 1 | | 27 | 3 | 1 | 9 | 0 | 0 | 0 | 0 | | | | | | Amsterdam_VUmc | 1.8 |
| AB_0_001 | sub-916007 | 2 | | 2 | | 28 | 3 | 2 | 11 | 0 | 0 | 0 | 0 | | | | | | Amsterdam_VUmc | 1.8 |
| AB_0_002 | sub-916011 | 2 | | 1 | | 33 | 3 | 2 | 15 | 0 | 0 | 0 | 0 | | | | | | Amsterdam_VUmc | 1.8 |
| AB_0_003 | sub-916013 | 2 | | 1 | | 26 | 3 | 1 | 18 | 0 | 0 | 0 | 0 | | | | | | Amsterdam_VUmc | 1.8 |
| AB_0_004 | sub-916015 | 2 | | 1 | | 47 | 3 | 1 | 11 | 0 | 0 | 0 | 0 | | | | | | Amsterdam_VUmc | 1.8 |

Table 1: Excerpt of the demographic table used in this study. Only a subset of subjects and variables is shown for illustrative purposes. The complete table contains 2126 subjects.

# .2 New pipeline results

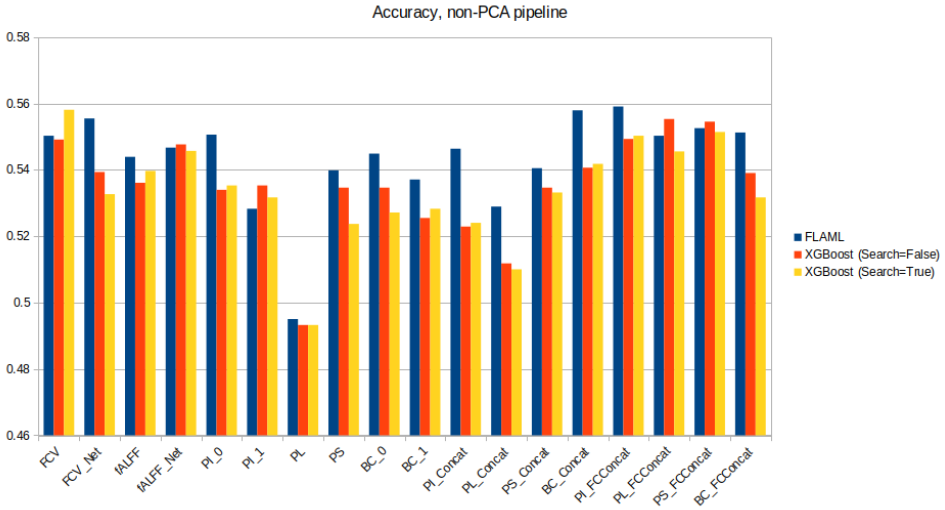



Figure 21: Accuracy visualization

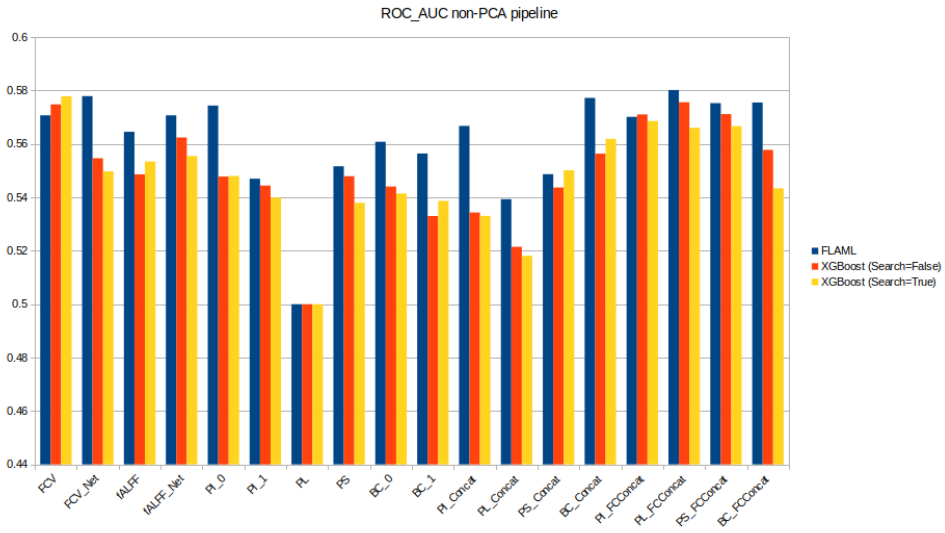



Figure 22: ROC_AUC visualization

# .3 Results comparison

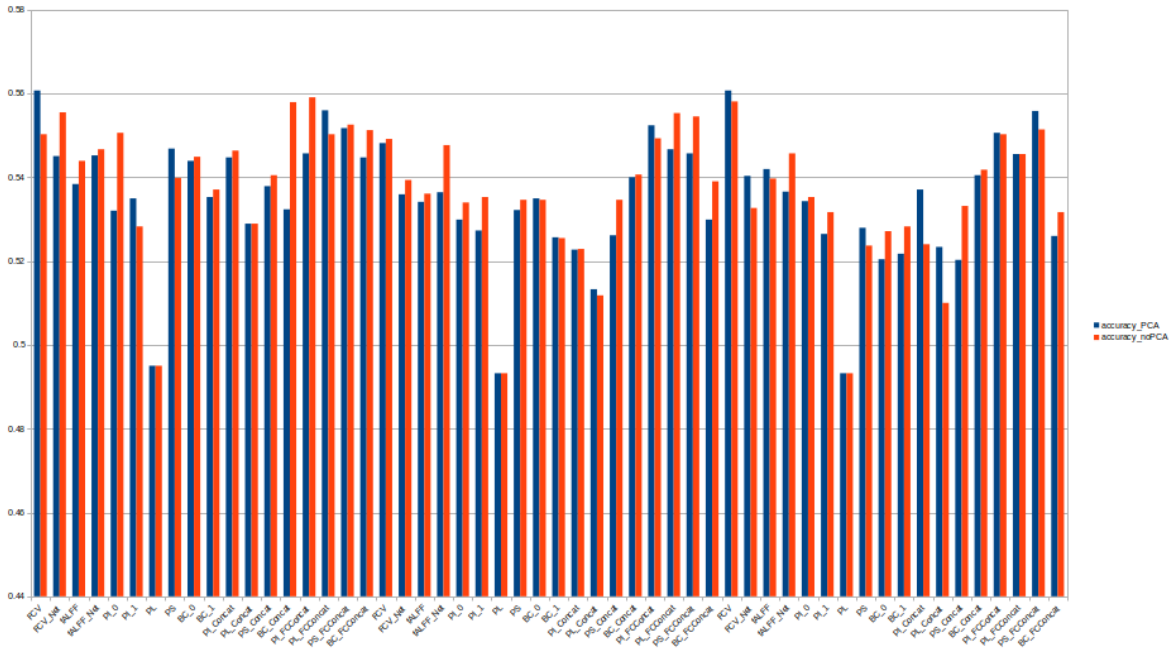



Figure 23: Accuracy values for each feature set; Blue being of the PCA model and red being of the non-PCA model. For each feature set we have in order: FLAML, XGBoost with search false and XGBoost with search true representation

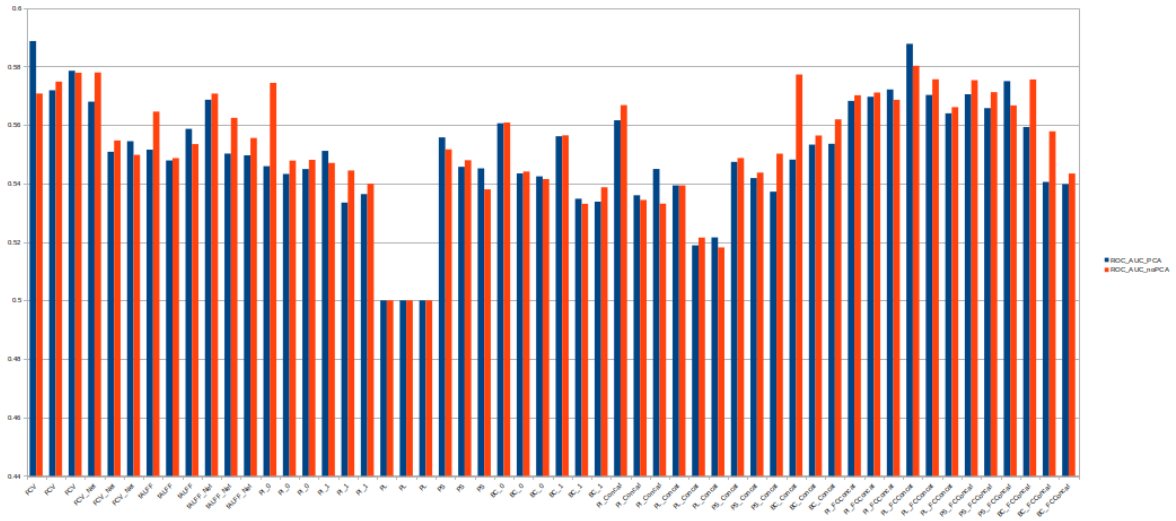



Figure 24: ROC_AUC values for each feature set; Blue being of the PCA model and red being of the non-PCA model. For each feature set we have in order: FLAML, XGBoost with search false and XGBoost with search true representation

# .4 Source

The pipelines used in this thesis can be found here (*https* : *//github.com/martavise/clean_thesis*).