Introduction to Topological Data Analysis
Part I: Topological Signatures from Point Clouds

Tom Needham
The Ohio State University

Cőde Bootcamp
The Ohio State University
May 28, 2019

Overview of Topological Data Analysis (TDA)

Idea

Use ideas from the mathematical field of algebraic topology to describe structure of a dataset

- Connected components (a.k.a. clustering)
- "Holes" of various dimensions ("generalized clustering")

Benefits

Descriptions are

- Multiscale — get pictures of the data at multiple resolutions.
- Stable — topology is insensitive to noise.
- Flexible — topological methods apply to all types of data, can be used to get many types of insights.

Applications of TDA

Example application domains:

- Biomedicine
  - Discovered new subgroup of breast cancer [Nicolau et. al., 2013]
  - Predicts survival time for brain cancer patients [Crawford et. al., 2016]
- Shape Analysis
  - Used to classify 3D shapes for computer vision applications [Chazal et. al., 2009]
  - Applied to classify anatomical surfaces [Turner et. al., 2013]
- Machine Learning
  - Used to analyze structure of neural networks [Rieck et. al., 2018]
  - Topological features improve performance of neural networks [Carrière et. al., 2019]

Motivation: Hierarchical Clustering

Clustering data is a basic task in unsupervised learning.

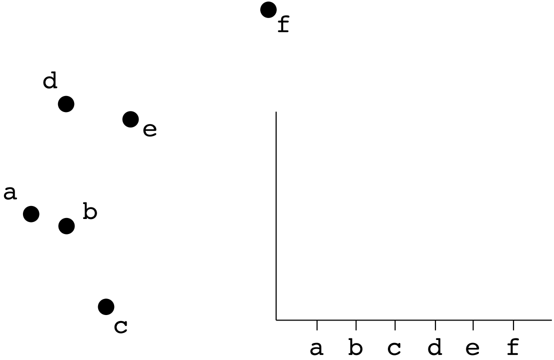
Methods like k-Means, DBSCAN, etc. partition data into clusters.

- Require parameter tuning.
- Produce one partition into clusters, potentially ignoring finer clustering structure.

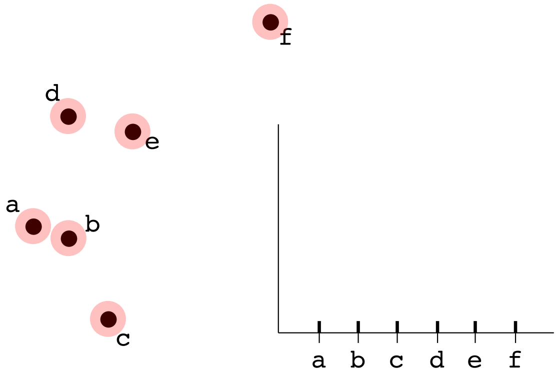
Hierarchical Clustering produces a multiscale summary of cluster structure, visualized as a dendrogram.

See Example 1 in accompanying Jupyter Notebook.

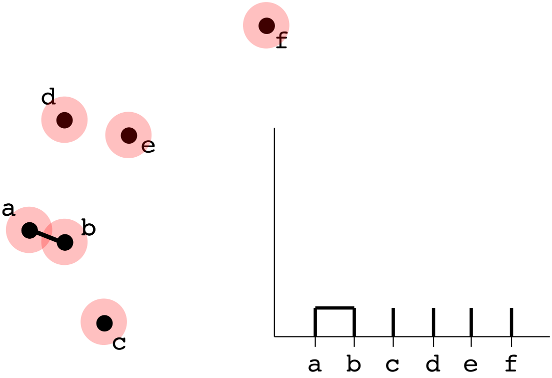
Example: Hierarchical Clustering



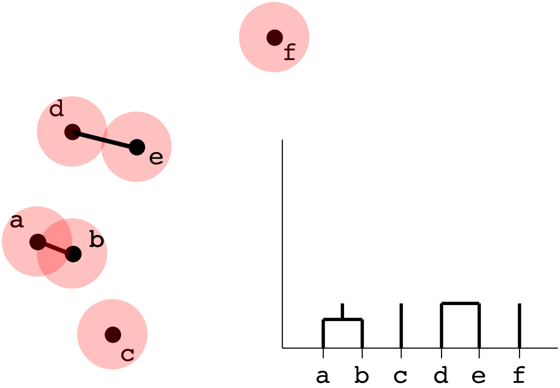
Example: Hierarchical Clustering



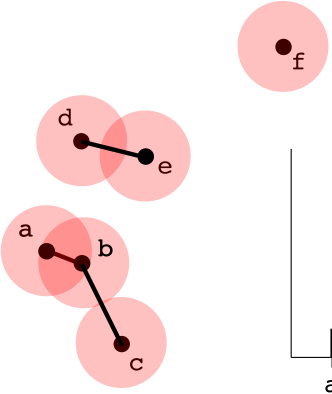
Example: Hierarchical Clustering

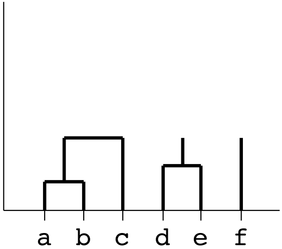


Example: Hierarchical Clustering

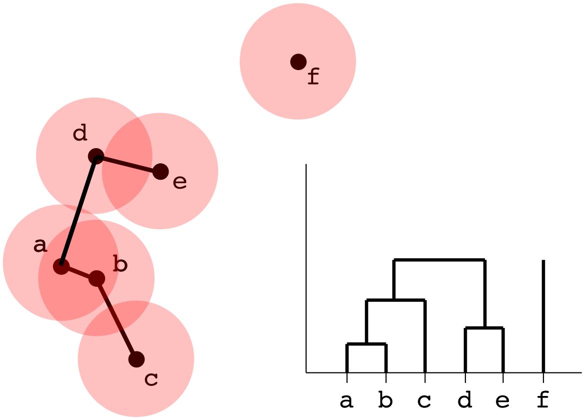


Example: Hierarchical Clustering

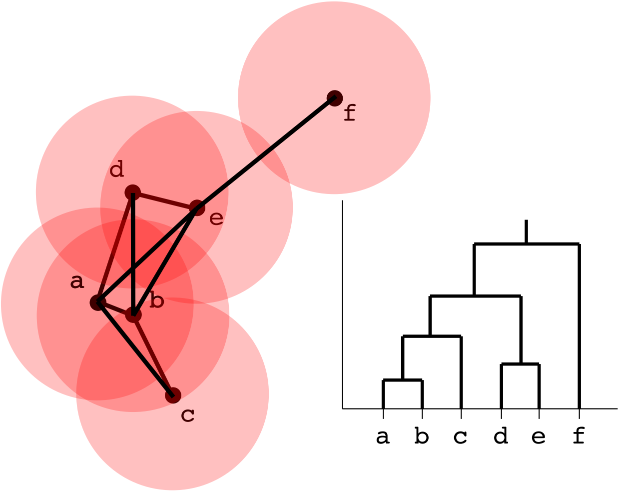




Example: Hierarchical Clustering



Example: Hierarchical Clustering



Concepts from Topology

Topology is a field of math which studies geometrical objects up to loose notions of "equivalence".

Each such object is called a (topological) space, denoted X.

Roughly, spaces are equivalent if one can be deformed into the other via stretching and bending, without creating or closing holes.

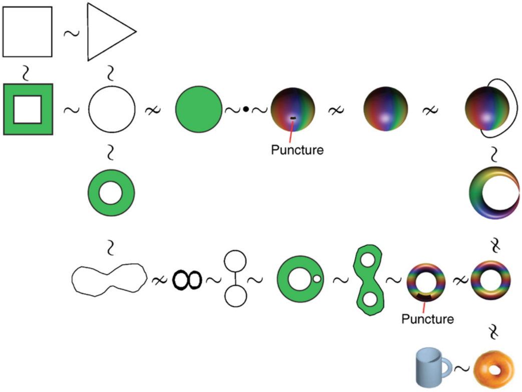


Figure: Homotopy equivalence, from Singh et. al. 2008.

Concepts from Topology

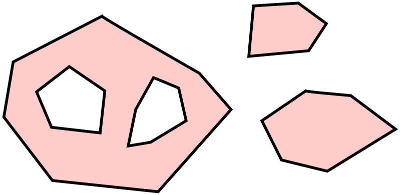
Algebraic topology is a subfield of topology where one computes invariants of a space which distinguish it from other spaces.

To each space X, we can associate a vector space $H_k(X)$ called the kth homology vector space of X.

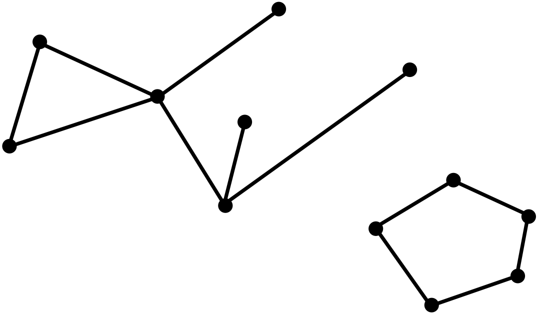
Its dimension $\beta_k(X)$ is called the kth Betti number of X.

The Betti number $\beta_k(X)$ counts "k-dimensional holes" in X:

- 0-dimensional — # of connected pieces
- 1-dimensional — # of unfilled loops
- 2-dimensional — # of unfilled "voids" (interior of a basketball)
- k-dimensional — well-defined concept we can't visualize



Examples of Betti Numbers



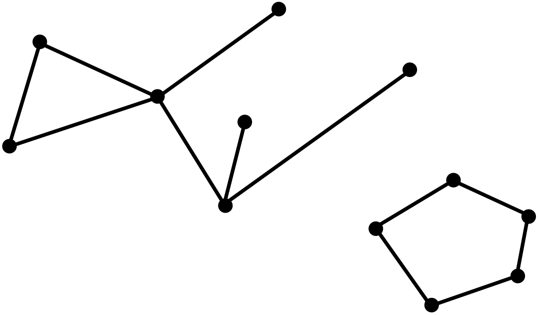
$\beta_0 =$

$\beta_1 =$

$\beta_2 =$

Figure: Disconnected graph.

Examples of Betti Numbers



$$\beta_0 = 2$$

$$\beta_1 = 1$$

$$\beta_2 = 0$$

Figure: Disconnected graph.

Examples of Betti Numbers

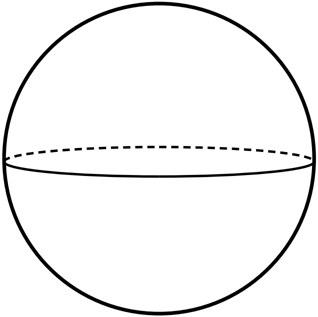


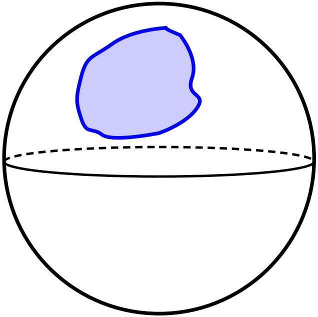
Figure: Surface of a sphere.

$\boldsymbol{\beta}_0 =$

$\boldsymbol{\beta}_1 =$

$\boldsymbol{\beta}_2 =$

Examples of Betti Numbers



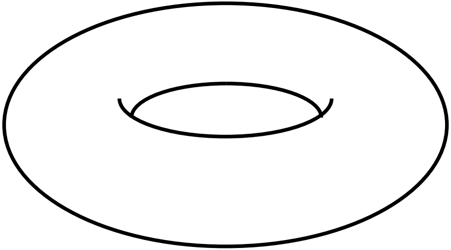
$\boldsymbol{\beta}_0 = 1$

$\boldsymbol{\beta}_1 = 0$

$\boldsymbol{\beta}_2 = 1$

Figure: Any loop on the sphere can be filled in with a disk.

Examples of Betti Numbers



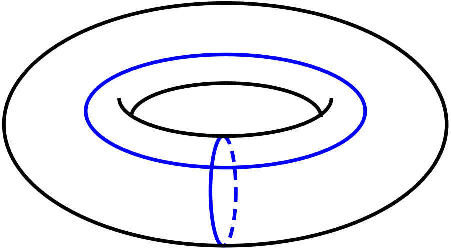
$\boldsymbol{\beta}_0 =$

$\boldsymbol{\beta}_1 =$

$\boldsymbol{\beta}_2 =$

Figure: Torus (surface of a donut).

Examples of Betti Numbers



$\beta_0 = 1$

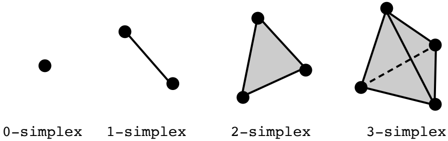
$\beta_1 = 2$

$\beta_2 = 1$

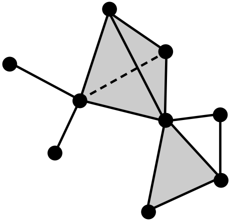
Figure: Blue loops can't be filled by disks that stay in the surface.

Simplicial Homology

A *k*-simplex is a *k*-dimensional generalization of a triangle.



A simplicial complex is a space obtained by gluing together simplices along lower-dimensional faces.

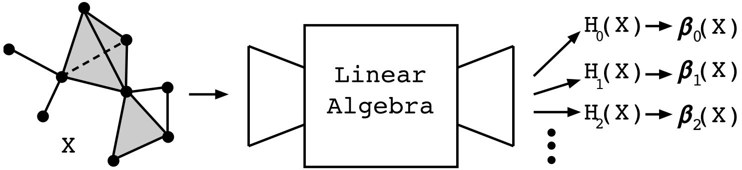


Simplicial Homology

Computing homology/Betti numbers of simplicial complexes is easy!

Boils down to linear algebra:

- Gluing process is described by linear maps.
- Homology is computed from kernels and images of these maps.



How Does This Apply to Data?

The most common type of data is a point cloud — a set of vectors $X = \{\vec{x}_1, \ldots, \vec{x}_N\}$, each $\vec{x}_j \in \mathbb{R}^d$.

This is a simplicial complex with only 0-dimensional simplices and no interesting topology; i.e.,

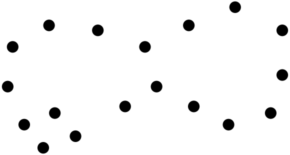
$$\beta_0 = N, \quad \beta_1, \beta_2, \ldots = 0.$$

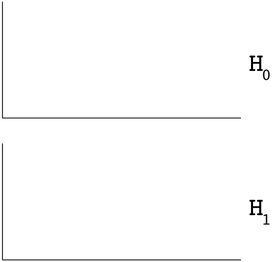
Idea

Construct a family of simplicial complexes following the example of hierarchical clustering.

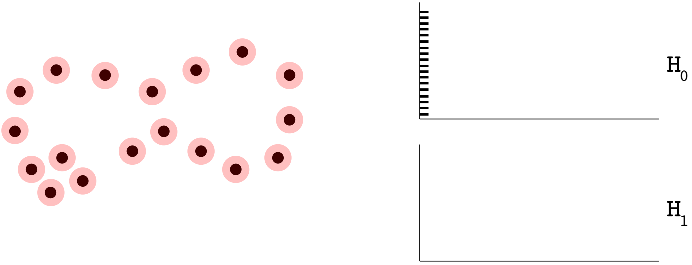
This leads to the main tool in TDA: persistent homology.

Persistent Homology - An Example

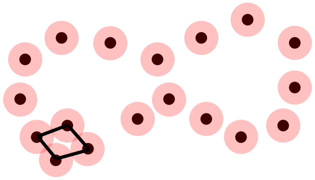


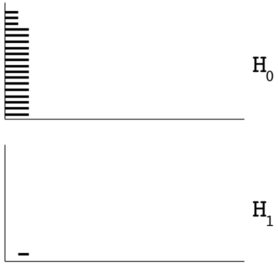


Persistent Homology - An Example

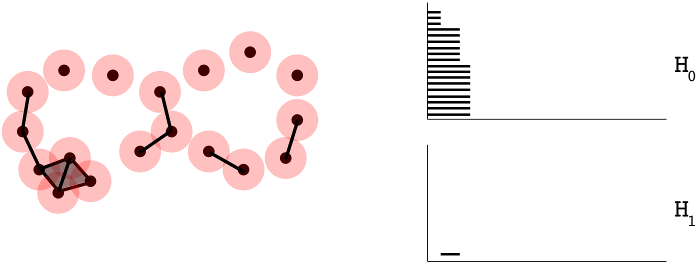


Persistent Homology - An Example

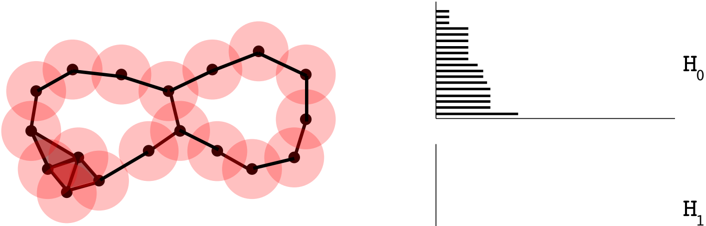




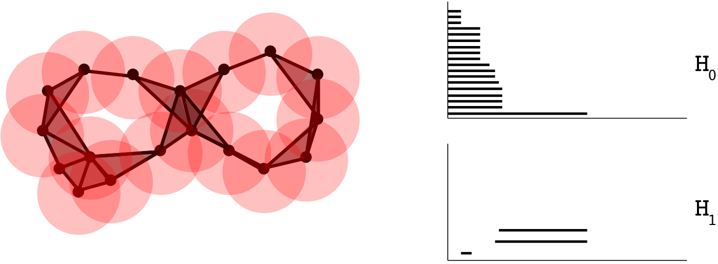
Persistent Homology - An Example



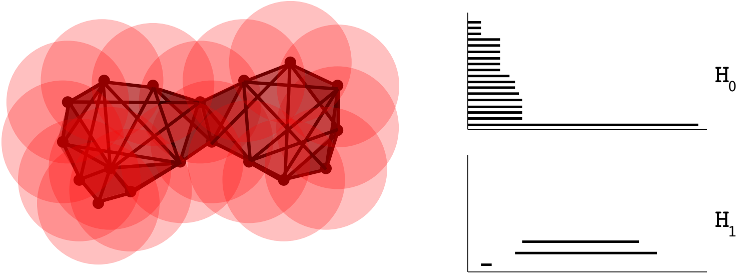
Persistent Homology - An Example



Persistent Homology - An Example

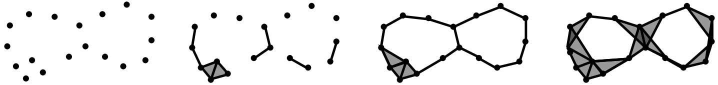


Persistent Homology - An Example



Terminology

Such a family is called a filtered simplicial complex.



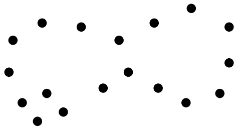
There are many techniques for creating them.

The previous example is called a Cech complex.

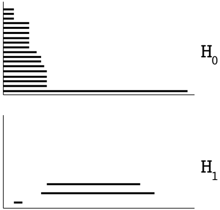
In computational examples, we'll use a related construction called a Vietoris-Rips complex.

Terminology

The topological signatures we get from persistent homology are called barcodes.



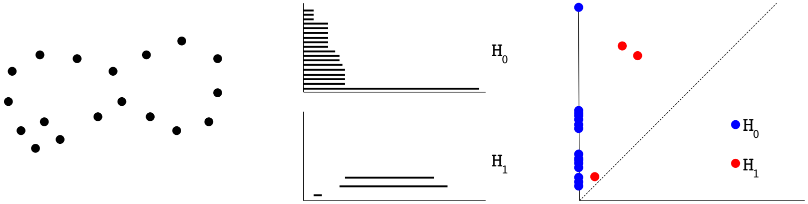
Dataset X



Barcodes for X

Terminology

We can record the "birth time" and "death time" of each topological feature to get a persistence diagram.

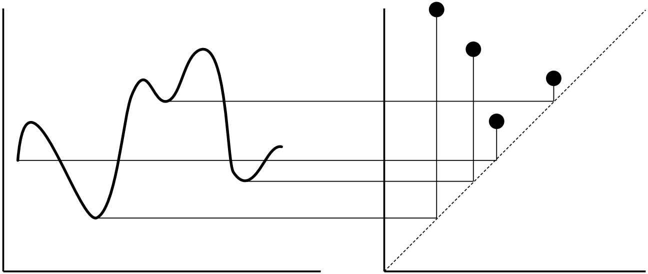


Dataset X   Barcodes for X   Persistence Diagram for X

See Example 2 in the accompanying Jupyter notebook.

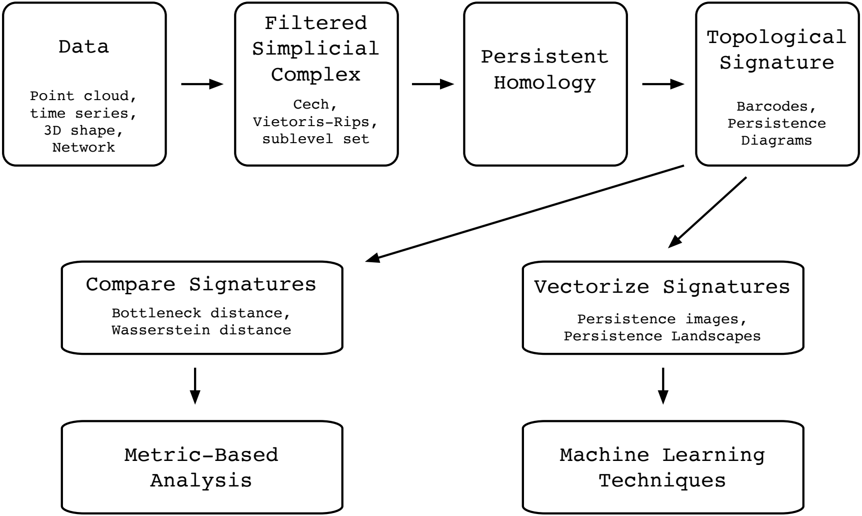
Terminology

Another common filtration is by sublevel sets of the graph of a function.



See Example 3 in the accompanying Jupyter notebook.

TDA Workflow



Next Time

- Comparing persistence diagrams via Bottleneck Distance
- Turning persistence diagrams into vectors for Machine Learning
- Applications:
  - Shape classification using persistence diagrams
  - Logistic regression on vectorized persistence diagrams