Topology Data Processing and Its Application

Zhiqi Li

Computer Science and Technology
Zhejiang University

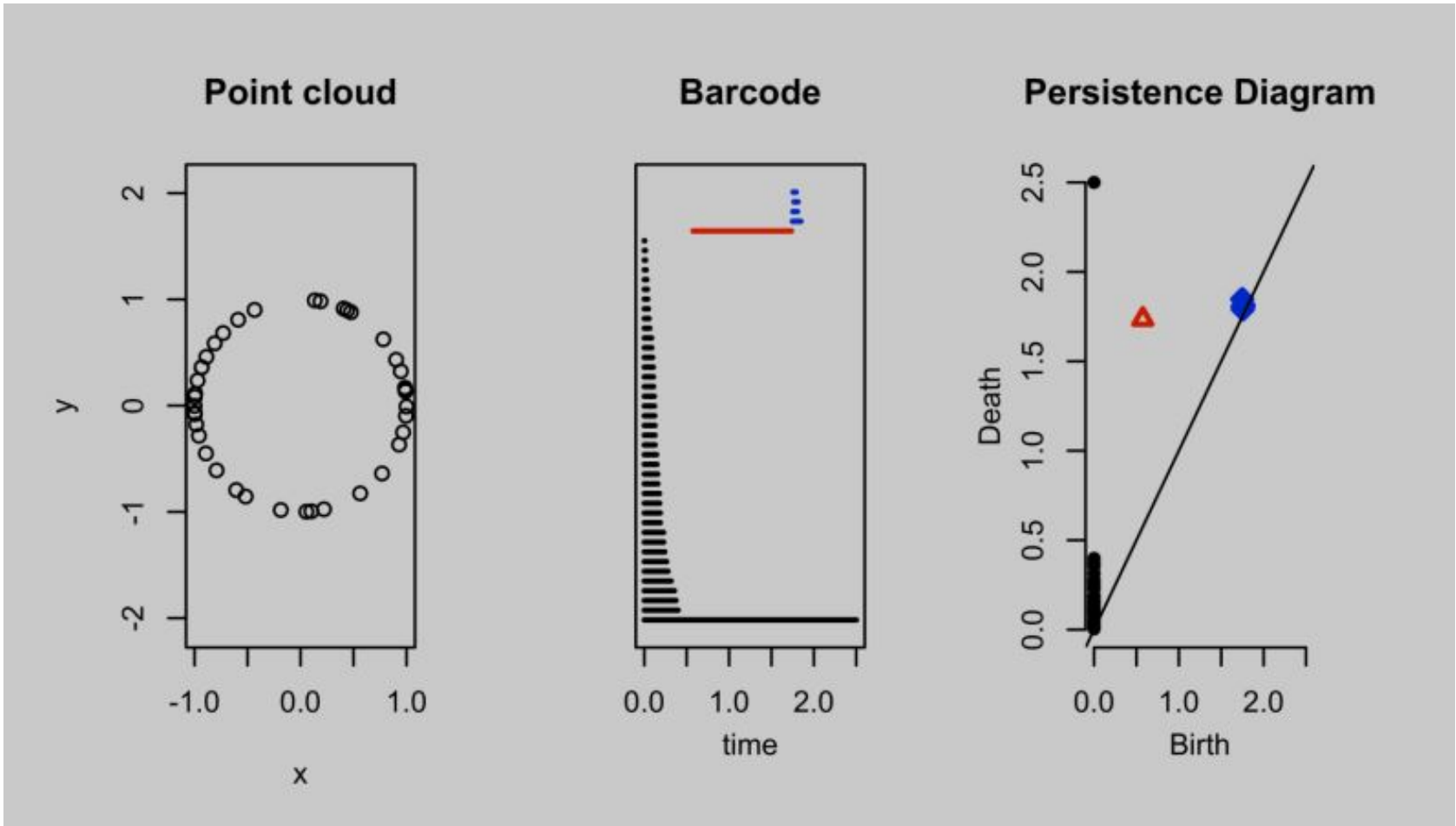
December 5, 2022

目录

Two methods, Persistence Homology and Mapper, are introduced here, which reduce high dimensional data sets into simplicial complexes with far fewer points which can capture topological and geometric information at a specified resolution. These two methods are in the form of simplicial complexes.

1. Construct a Complex on the point cloud with a paragram ϵ
2. Increase ϵ and update and record the topology features
3. Draw the Barcode and Persistence Diagram.

Simple Example for Persistence Homology



Algorithm Structure for Mapper

Given a number N of intervals and p, a percent overlap

1. Calculate a filter function of each point

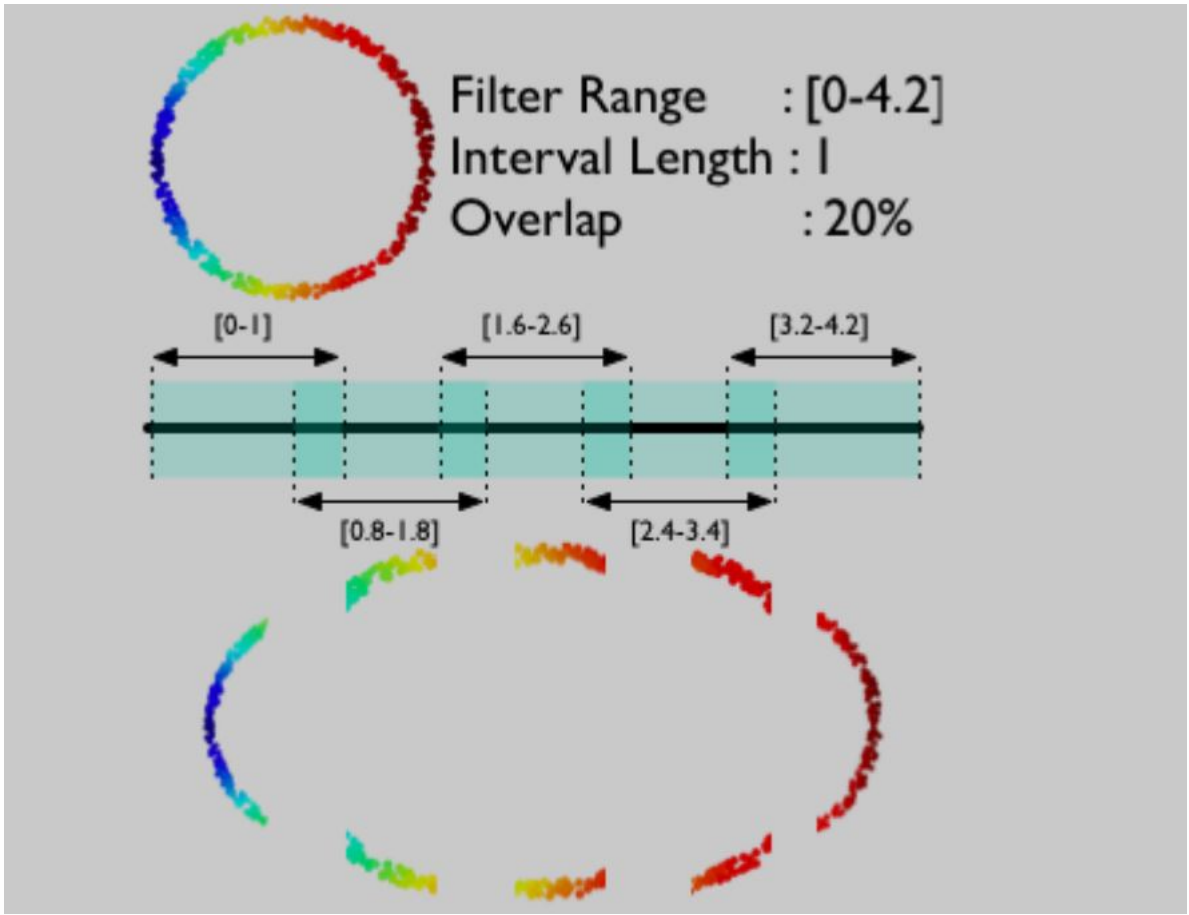
2. Separate points into N bins based on the value of the filter function, the bins should have p overlap with its adjacent interval

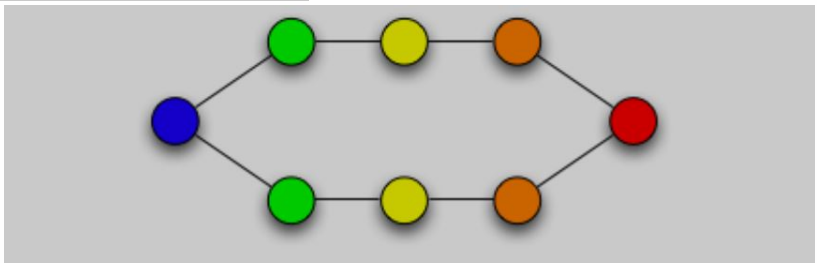
3. Select a clustering scheme and apply it to each bin

4. Draw a graph with the clustering results of 3 as the vertex. If two clusters(even in different bins) have not-empty adjacent, connect the two vertexes.

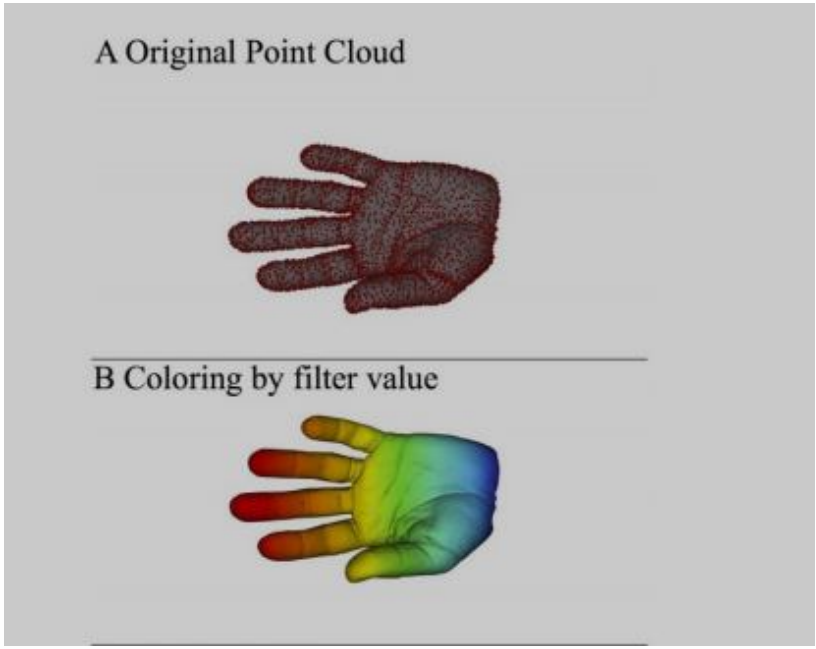
Key: filter function, cluster algorithm

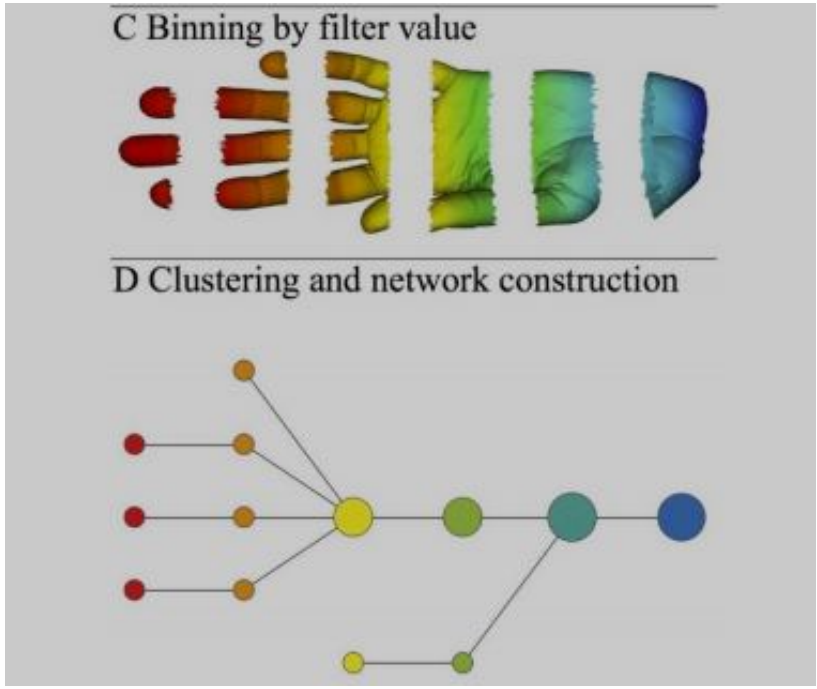
Simple Example for Mapper





Simple Example for Mapper





目录

1. Linearly independent points:

   points $v_0, ..., v_k \in R^n$ is linearly independent $\triangleq$

   vectors $v_0v_1, ..., v_0v_k \in R^n$ are linearly independent

2. Simplex

Let $v_0, ..., v_k$ be linearly independent points in R^n. The k-simplex denoted by $[v_0, ..., v_k]$ is the topological space given by the set

$$\{\sum_{i=0}^{k} t_i v_i | t_0 + t_1 + ... + t_k = 1, t_i \geq 0\} \tag{1}$$

with the topology induced by the Euclidian metric. The numbers t_i are the coordinates of the point $x = \sum_i t_i v_i \in [v_0, ..., v_k]$

Example of simplex

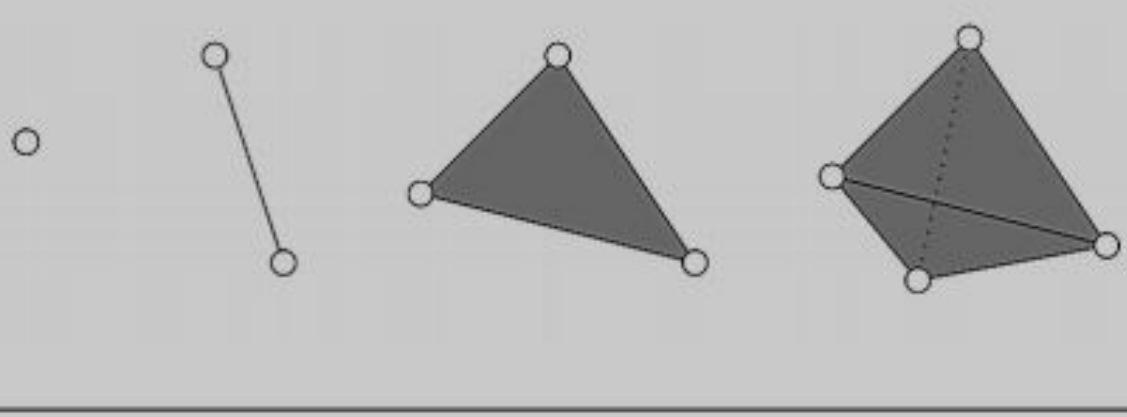


FIGURE 2.2: From left to right respectively are Simplices of dimension zero, one, two and three.

1. Simplicial Complex

Set X be a finite set. A simplicial complex K on X is a set of subsets of X such that:

$$1.\{x\} \in K \text{ for any x in X} \tag{2}$$

$$2.\text{if } \sigma \in \text{ K and } \tau \subseteq \sigma, \text{ then } \tau \in K \tag{3}$$

2. Notation K_i

We use the symbol K_i to denote the set of simplices in K of dimension i, for example $K_0 = X$.

3. Notation $d_i\sigma$

For any n-dimensional simplex $\sigma = \{x_0 < x_1 < ... < x_n\} \in K_n$ and $0 \leq i \leq n$, define $d_i\sigma$ to be the (n−1)-dimensional simplex in K given by omitting x_i from σ.

1. R-module

Suppose that R is a ring, and 1 is its multiplicative identity. A left R-module M consists of an abelian group (M, +) and an operation $\cdot : R \times M \to M$ such that for all r, s in R and x, y in M, we have :

$$r \cdot (x + y) = r \cdot x + r \cdot y \tag{4}$$

$$(r + s)\, x = r\, x + s\, x (r + s) \cdot x = r \cdot x + s \cdot x \tag{5}$$

$$(rs)\, x = r\, (s\, x) \tag{6}$$

$$(rs) \cdot x = r \cdot (s \cdot x) 1\, x = x. 1 \cdot x = x. \tag{7}$$

One may write ${}_R M$ to emphasize that M is a left R-module. A right R-module M_R is defined similarly in terms of an operation $\cdot : M \times R \to M$.

1. A free module is a module with a basis and acts as a vector space with coefficient on R.

2. Basis
For a ring R and an R-module M, the set E is a basis for M if:

(1) E is a generating set for M; that is to say, every element of M is a finite sum of elements of E multiplied by coefficients in R;

(2) E is linearly independent, that is, for every subset $\{e_1, e_2, ..., e_n\}$ of distinct elements of E, $r_1e_1 + r_2e_2 + \cdots + r_ne_n = 0_M$ implies that $r_1 = r_2 = \cdots = r_n = 0_R$ (where 0_M is the zero element of M and 0_R is the zero element of R).

3. Given a set E and ring R, there is a free R-module that has E as a basis: namely, the direct sum of copies of R indexed by E:

$$R^{(E)} = \bigoplus_{e \in E} R. \tag{8}$$

1. Notation: ∂_n

Let R be a commutative ring, and X be a set and K be a simplicial complex on X. Define $C_n(K) := \bigoplus_{\sigma \in K_n} R$. ∂_n is a map between free R-module $C_n(K)$ and $C_{n-1}(K)$:

$$\partial_n(e_\sigma) = e_{d_0\sigma} - e_{d_1\sigma} + ... + (-1)^n e_{d_n\sigma} \tag{9}$$

where e_σ is the generator of the components of R in $\bigoplus_{\sigma \in K_n} R$ indexed by $\sigma \in K_n$

We could give a sequence of homeomorphisms:

$$\cdots \xrightarrow{\partial_{n+1}} C_n(K) \xrightarrow{\partial_n} C_{n-1}(K) \to \cdots \to C_1(K) \xrightarrow{\partial_1} C_0(K) \to 0 \tag{10}$$

2. Property of ∂_n

1. the composition $\partial_n \circ \partial_{n+1}$ is trivial, which means that $Im\partial_{n+1} \subseteq Ker\partial_n$. And the elements of $Im_{\partial n+1}$ are called the boundaries and the elements of $Ker\partial_n$ are called cycles. Examples:

$$[v_1, v_2, v_3] \in C_2(K), \partial_2([v_1, v_2, v_3]) = [v_1, v_2] + [v_2, v_3] - [v_1, v_3] \Rightarrow$$
$$\partial_1([v_1, v_2] + [v_2, v_3] - [v_1, v_3]) = (v_1 - v_2) + (v_2 - v_3) - (v_1 - v_3) = 0$$

1. Homology group $\mathrm{H_n(K,R)}$
The quotient $\mathrm{Ker}\partial_n/\mathrm{Im}\partial_{n+1}$ is called the homology group of K with coefficients in R and is denoted by $\mathrm{H_n(K,R)}$. The isomorphism type of the R-module $\mathrm{H_n(K,R)}$ does not depend on the ordering we have chosen on X. For our purposes we are mainly interested in the case R being a finite field $\mathrm{F_p}$.

2. n-th Betti number $\beta_n(\mathrm{K,R})$
The dimension of the R-vector space denoted by $\mathrm{H_n(K,R)}$ is called the n-th Betti number of K with respect to R and is denoted by $\beta_n(\mathrm{K,R})$.

1. Why Homology group and Betti number important?

The homology $H_n(K, R)$ is an algebraic representation of certain topological features that X possesses. Its dimension (which is called Betti numbers) represents a count of those features. For example, $\beta_0(K, R)$ is the number of connected components of K, $\beta_1(K, R)$ is the number of certain loops and $\beta_2(K, R)$ of certain voids.

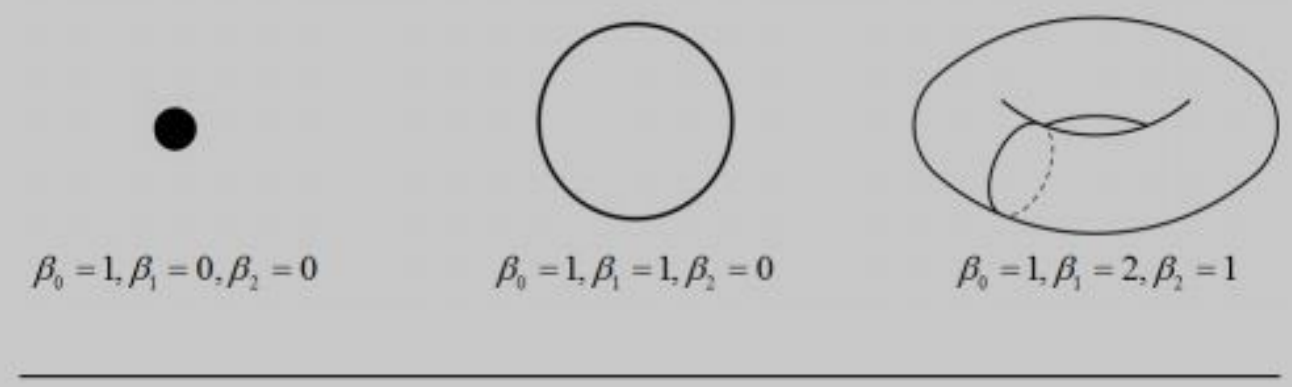


FIGURE 2.4: The first three Betti numbers for a point, a circle and a torus.

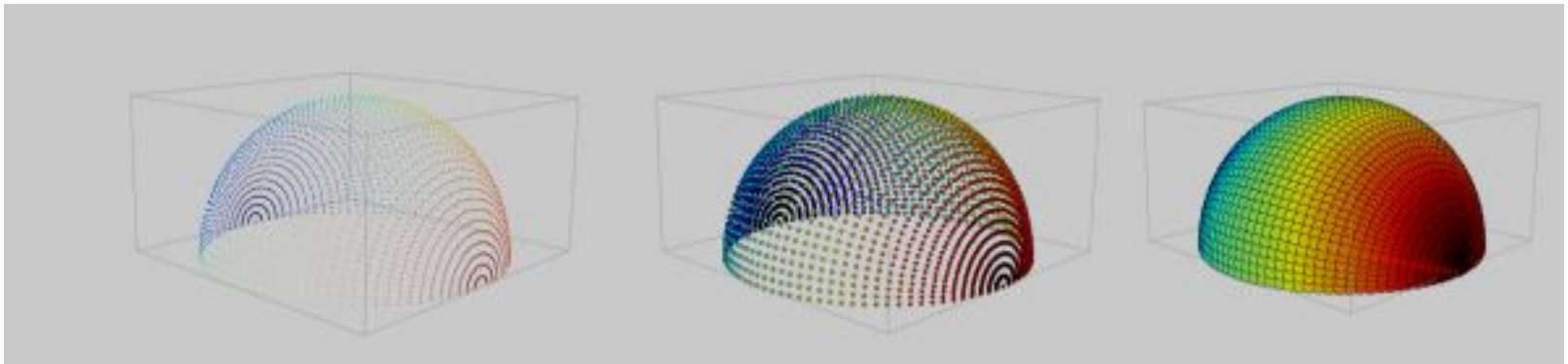
1. Purpose of Persistent Homology

(1) In the world of data we, unfortunately, do not have a direct access to a topological space or to a simplicial complex and we only have access to its sampling.

(2) Persistence is going to help us extend homology to the world of data and remarkably allows us to recover homological features such as holes in the space out of the sampling.

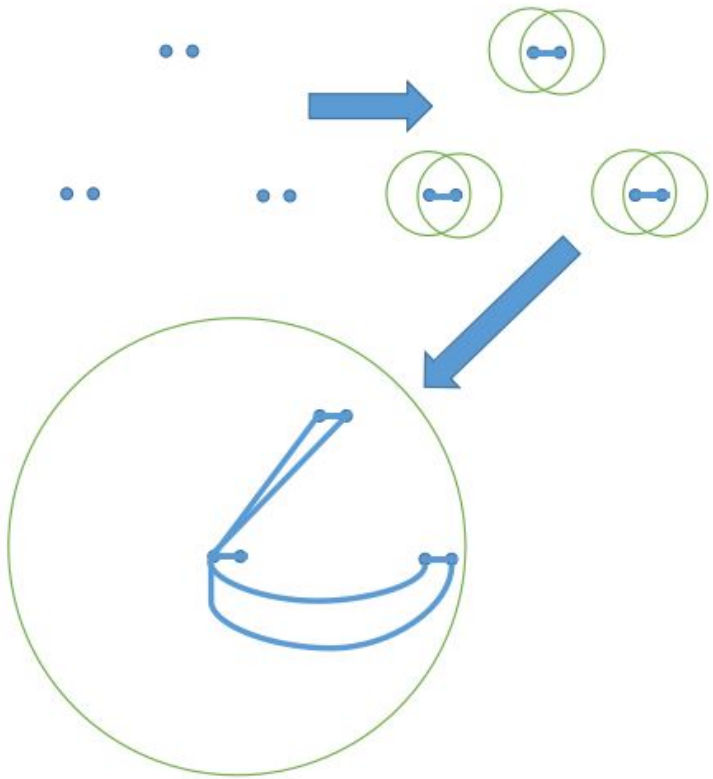
2. Basic Idea(1)

The basic approach of persistent homology is to sum up balls of size $\epsilon > 0$ around each data point of the sampling, then calculate the homology of the obtained space or simplicial complex and finally record how this homology is changing while varying $\epsilon > 0$.



1. Basic Idea(2)

In the last page, we mention "sum up balls of size $\epsilon > 0$" along with the increasing of ϵ. The actual meaning here is to add some connections around each data to the total simplicial complex. That is, construct a more "complex" simplicial complex along with the increasing of ϵ.



On the last page, we mention that the idea of persistent homology is to construct a more simplicial complex along with the increasing of ϵ and here are three alternatives of the complex.

(1) Vietoris-Rips complex
Define $\mathrm{V}(\mathrm{X}, \epsilon)$ to be a simplicial complex on the set X consisting of these subsets $\sigma \subseteq$ Xwhere $\mathrm{d}(\mathrm{x}, \mathrm{y}) < \epsilon$ for any $\mathrm{x}, \mathrm{y}\ \sigma$

(2) Cech complex
Define $\mathrm{C}(\mathrm{X}, \epsilon)$ to be a simplicial complex on the set X consisting of these subsets $\sigma \subseteq \mathrm{X}$ for which there is $\mathrm{y} \in \mathrm{X}$ such that $\mathrm{d}(\mathrm{x}, \mathrm{y}) < \epsilon$ for any $\mathrm{x} \in \sigma$

(3) Witness versions(save both computational time and memory)
Choose a subset $\mathrm{X_w} \subseteq \mathrm{X}$ of witnesses of X. Define $\mathrm{V_w}(\mathrm{X}, \epsilon)$, the witness version of the Vietoris-Rips complex, to be a complex on $\mathrm{X_w}$ that consists of these $\sigma \subseteq \mathrm{X_w}$ where for any $\mathrm{x}, \mathrm{y} \in \sigma$, there is $\mathrm{z} \in \mathrm{X}$ such that $\mathrm{d}(\mathrm{x}, \mathrm{z}) < \epsilon$ and $\mathrm{d}(\mathrm{y}, \mathrm{z}) < \epsilon$. Similarly define $\mathrm{C_w}(\mathrm{X}, \epsilon)$.

1. Persistence diagrams
Choose X_ϵ are complex we mention the last page. Then for any increasing sequence of real numbers $\epsilon_1 < \epsilon_2 < ... < \epsilon_i, ...$, then we get a sequence:

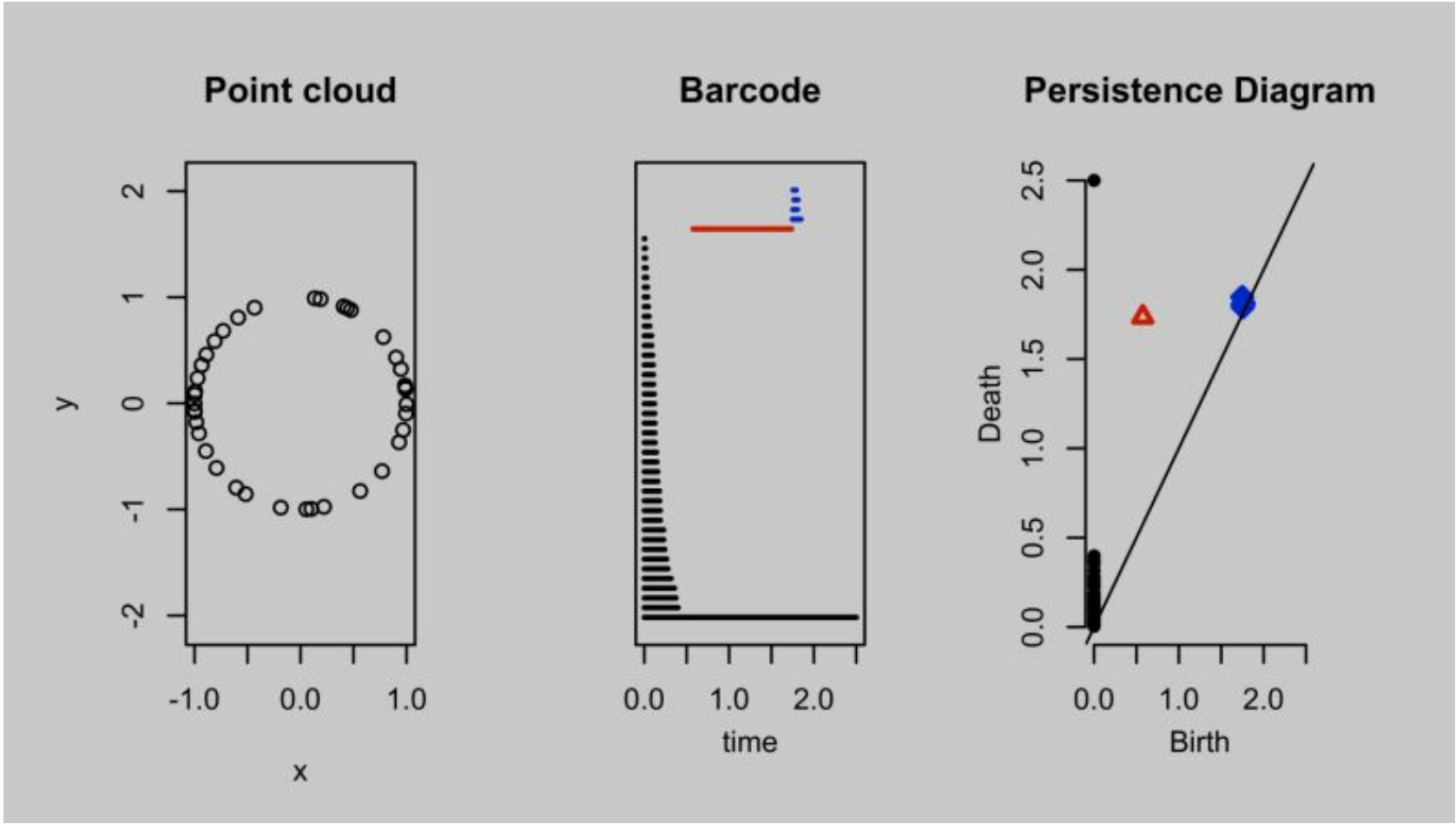
$$H_n(X_{\varepsilon_1}, R) \to H_n(X_{\varepsilon_2}, R) \to \cdots \to H_n(X_{\varepsilon_i}, R) \to \cdots \tag{11}$$

This sequence is called persistence diagrams or modules

2. Barcode
The bars in a barcode represent the lifetime of features across filtration i.e. tracking when a feature is born and when it dies

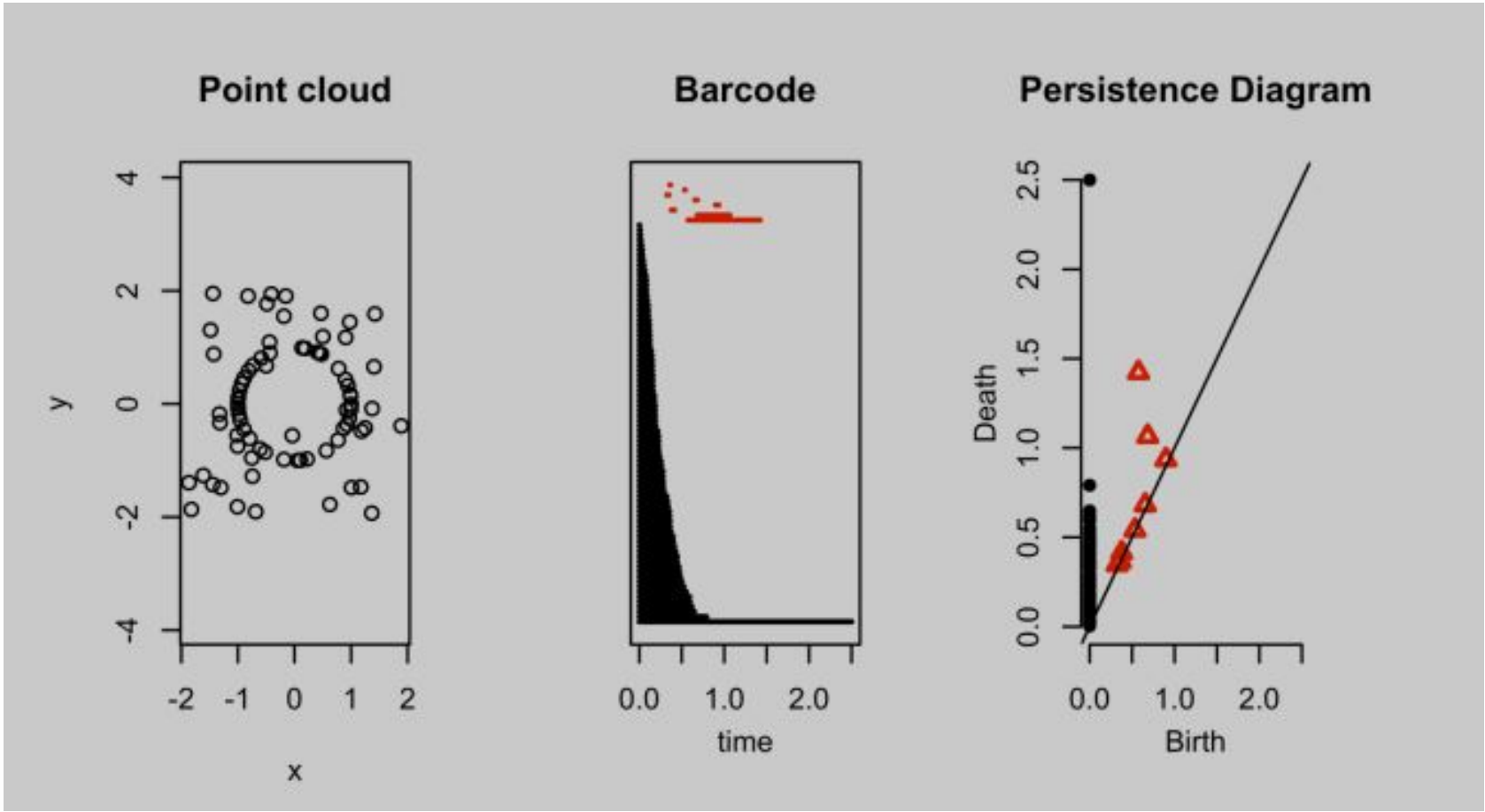
3. Persistence Diagram
Represent the lifetime in another way.

Example of Persistent Homology



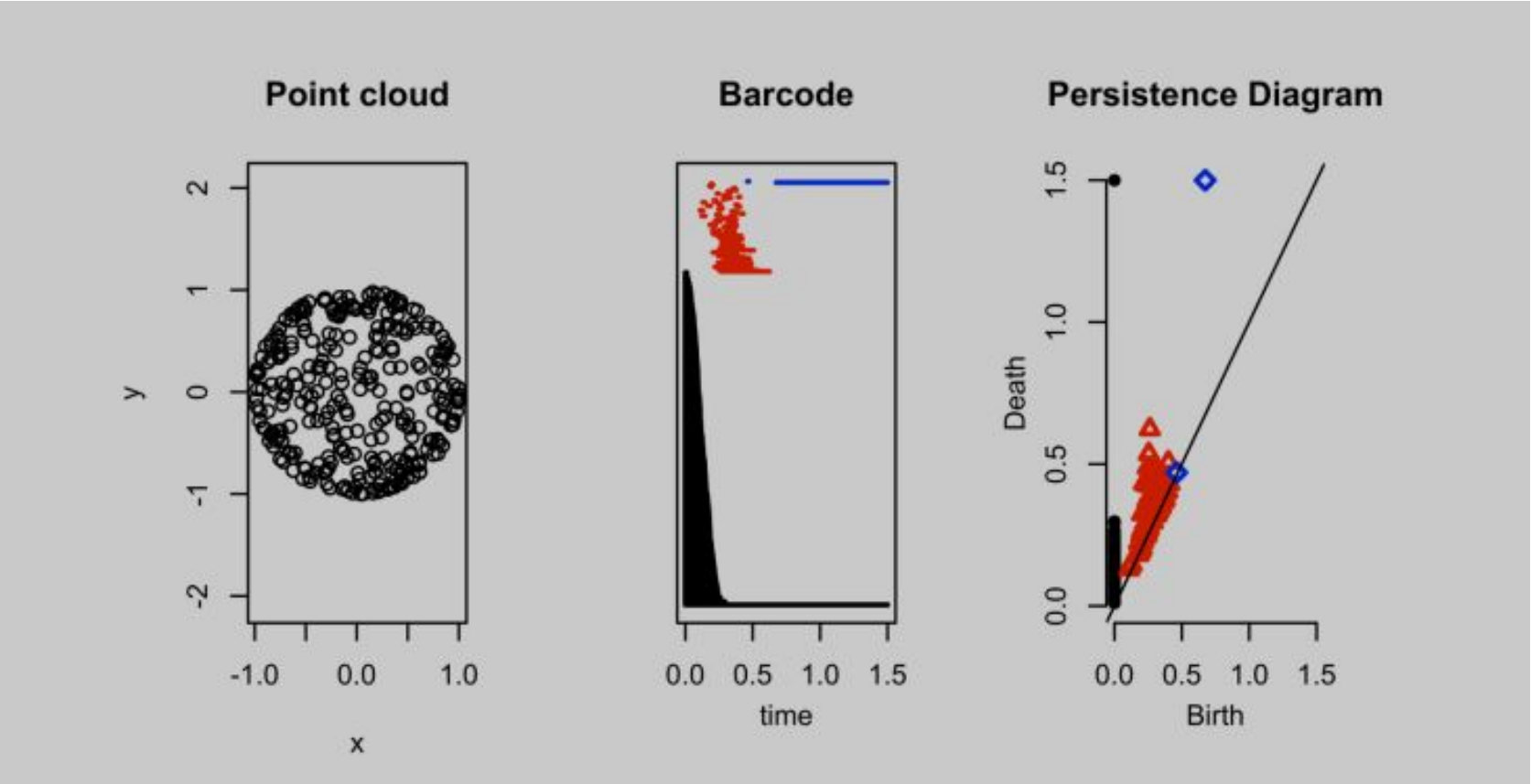
black bars represent the 0-dimensional cycles or vertices, a red bar represents a one-dimensional cycle(loop), and the blue bars represent cycles in H2(trapped voids) We like the feature that last for a long time.

Example of Persistent Homology



black bars represent the 0-dimensional cycles or vertices, a red bar represents a one-dimensional cycle(loop), and the blue bars represent cycles in H2(trapped voids) We like the feature that last for a long time.

Example of Persistent Homology



black bars represent the 0-dimensional cycles or vertices, a red bar represents a one-dimensional cycle(loop), and the blue bars represent cycles in H2(trapped voids) We like the feature that last for a long time.

I will discuss these along with Mapper later.

目录

Mapper is another tool in Topology Data Analysis.

1. Simplify Data
This method extract simple descriptions of high dimensional data sets in the form of simplicial complexes.

2. Non-Sensitive for Noise
mostly point clouds sampled from shapes of known topology can be identified and visualized even though in some cases noise is being there.

3. Find the Shape or Topology
This method produces a combinatorial object (a simplicial complex), whose interconnections reflect some aspects of the metric structure(Shape). So no coordinate!

4. A Framework
Use clustering method but doesn't depend on any clustering method.

5. Multiresolution
Recall the increasing of ϵ of persistent homology.

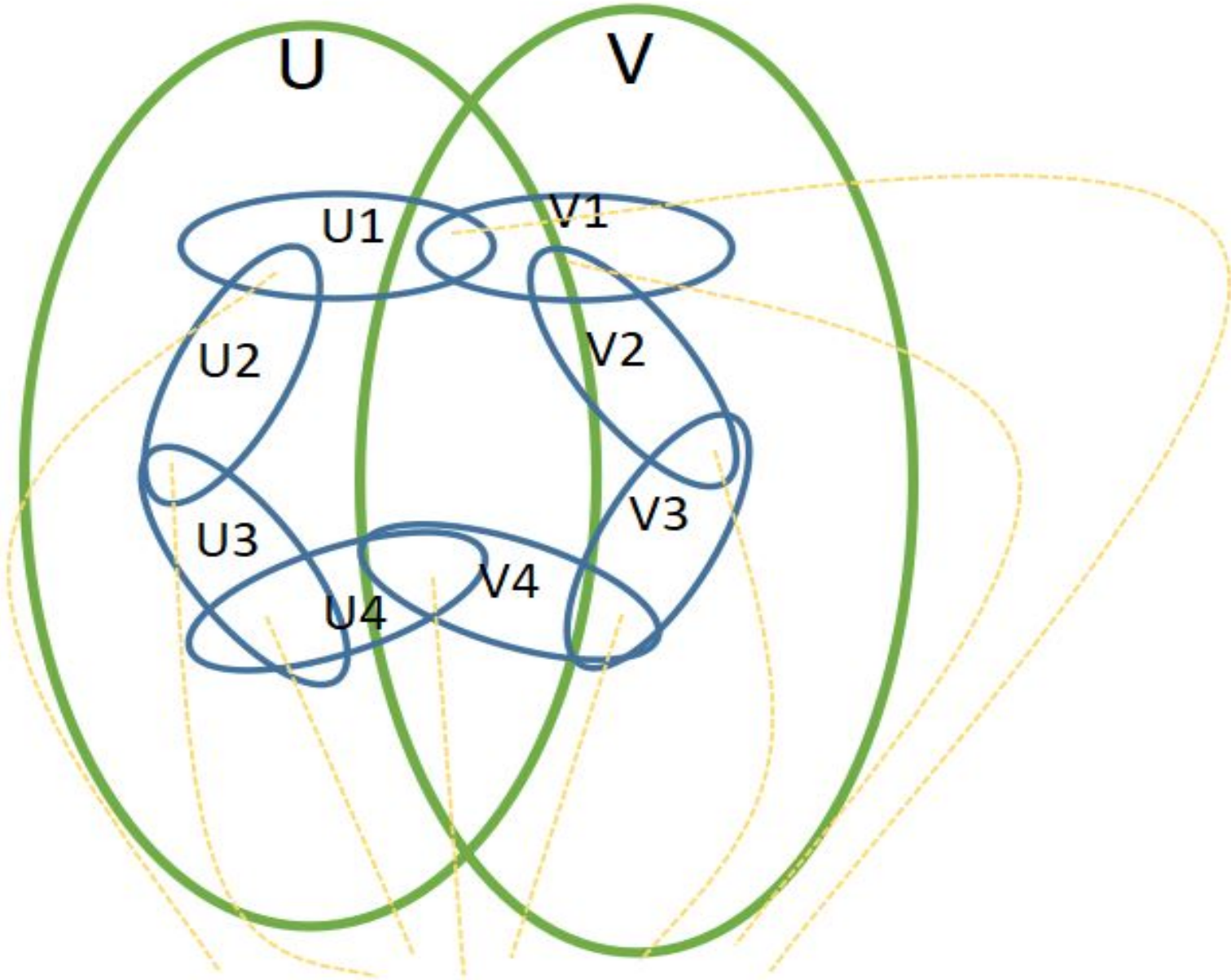
What's the difference between Mapper and Persistence Homology?

The simplicial complex.

The basic idea can be referred to as partial clustering, in that a key step is to apply standard clustering algorithms to subsets of the original data set, and then to understand the interaction of the partial clusters formed in this way with each other.

For example, given two sets U and V, which satisfy $U \cap V \neq \emptyset$. Then apply cluster algorithm to the set U and V, and get the clusters $\{U_i\}_i$ for U and clusters $\{V_j\}_j$ for V, which are called the partial clusters. If $U_i \cap V_j$, we can say they have a interaction. Mapper method is constructed on these interactions.

Basic Idea



All the overlap areas are interactions.

Although the interest in this construction comes from applying it to point cloud data and functions on point cloud data, it is motivated by well known constructions in topology. In the interest of clarity, we will introduce this theoretical construction first, and then proceed to develop the analogous construction for point cloud data. We will refer to the theoretical construction as the topological version and to the point cloud analogue as the statistical version.

In this part, we talk about the simplicial complex of mapper–Nerve.

1. A partition of unity

A partition of unity subordinate to the finite open covering U is a family of real valued functions $\{\phi_{\alpha\in \mathrm{A}}\}_{\alpha\in \mathrm{A}}$ with the following properties:

(1) $0 \leq \phi_\alpha(\mathrm{x}) \leq 1$ for all $\alpha \in \mathrm{A}$ and $\mathrm{x} \in \mathrm{X}$.

(2) $\sum \alpha\phi_\alpha(\mathrm{x}) = 1$ for all $\mathrm{x} \in \mathrm{X}$.

(3) The closure of the set $\{\mathrm{x} \in \mathrm{X} | \phi_\alpha(\mathrm{x}) > 0\}$ is contained in the open set U_α.

2. Nerve of the covering

Given a finite covering $\mathrm{U} = \{\mathrm{U}_\alpha\}\alpha \in \mathrm{A}$ of a space X, we define the nerve of the covering U to be the simplicial complex N(U) whose vertex set is the indexing set A, and where a family $\{\alpha_0, \alpha_1, ..., \alpha_\mathrm{k}\}$ spans a k-simplex in N(U) if and only if $\mathrm{U}_{\alpha_0} \cap \mathrm{U}_{\alpha_1} \cap ... \cap \mathrm{U}_{\alpha_\mathrm{k}} \neq \emptyset$

1. Barycentric Coordinates for Simplex
If $\{v_0, v_1, ..., vk\}$ are the vertices of a simplex, then the points v in the simplex correspond in a one-to-one and onto way to the set of ordered k-tuples of real numbers $(r_0, r_1, ..., r_k)$ which satisfy
$0 \leq r_i \leq 1 \text{and} \sum_{i=0}^{k} r_i = 1$. Here $(r_0, r_1, ..., r_k)$ is the barycentric coordinatization of simplex

2. Barycentric Coordinatization of N(U)
For any point $x \in X$, we let $T(x) \subseteq A$ be the set of all α so that $x \in U_\alpha$. We now define $\rho(x)$ N(U) to be the point in the simplex spanned by the vertices $\alpha \in T(x)$, whose barycentric coordinates are $(\phi_{\alpha_0}(x), \phi_{\alpha_1}(x), ..., \phi_{\alpha_l}(x))$, where $\{\alpha_0, \alpha_1, ..., \alpha_l\}$ is an enumeration of the set T(x). The map can easily be checked to be continuous, and provides a kind of partial coordinatization of X, with values in the simplicial complex N(U)

Now suppose that we are given a space equipped with a continuous map $f : X \to Z$ to a parameter space Z, and that the space Z is equipped with a covering $U = \{U_\alpha\}_{\alpha \in A}$, again for some finite indexing set A. Since f is continuous, the sets $f^{-1}(U_\alpha)$ also form an open covering of X. For each α, we can now consider the decomposition of $f^{-1}(U_\alpha)$ into its path connected components, so we write $f^{-1}(U_\alpha) = \cup_{i=1}^{j_\alpha} V(\alpha, i)$, where j_α is the number of connected components in $f^{-1}(U_\alpha)$. We write U for the covering of X obtained this way from the covering U of Z.

1. Map between Coverings

If we have two coverings $U = \{U_\alpha\}_{\alpha \in A}$ and $V = \{V_\beta\}_{\beta \in B}$ of a space X, a map of coverings from U to V is a function $f : A \to B$ so that for all $\in A$, we have $U_\alpha \subseteq V_{f(\alpha)}$ for all $\alpha \in A$.

2. Induced Map between Simplicial Complex

If we are given a map of coverings from $U = \{U_\alpha\}_{\alpha \in A}$ to $V = \{V_\beta\}_{\beta \in B}$, i.e. a map of sets $f : A \to B$ satisfying the conditions above, there is an induced map of simplicial complexes $N(f) : N(U) \to N(V)$, given on vertices by the map f. (Note: $U_\alpha \subseteq V_{f(\alpha)}$)

3. Multiresolution Structure If we have a family of coverings $U_i, i = 0, 1, ..., n$, and maps of coverings $f_i : U_i \to U_{i+1}$ for each i, we obtain a diagram of simplicial complexes and simplicial maps:

$$N(\mathcal{U}_0) \overset{N(f_0)}{\to} N(\mathcal{U}_1) \overset{N(f_1)}{\to} \dots \overset{N(f_{n-1})}{\to} N(\mathcal{U}_N) \quad (12)$$

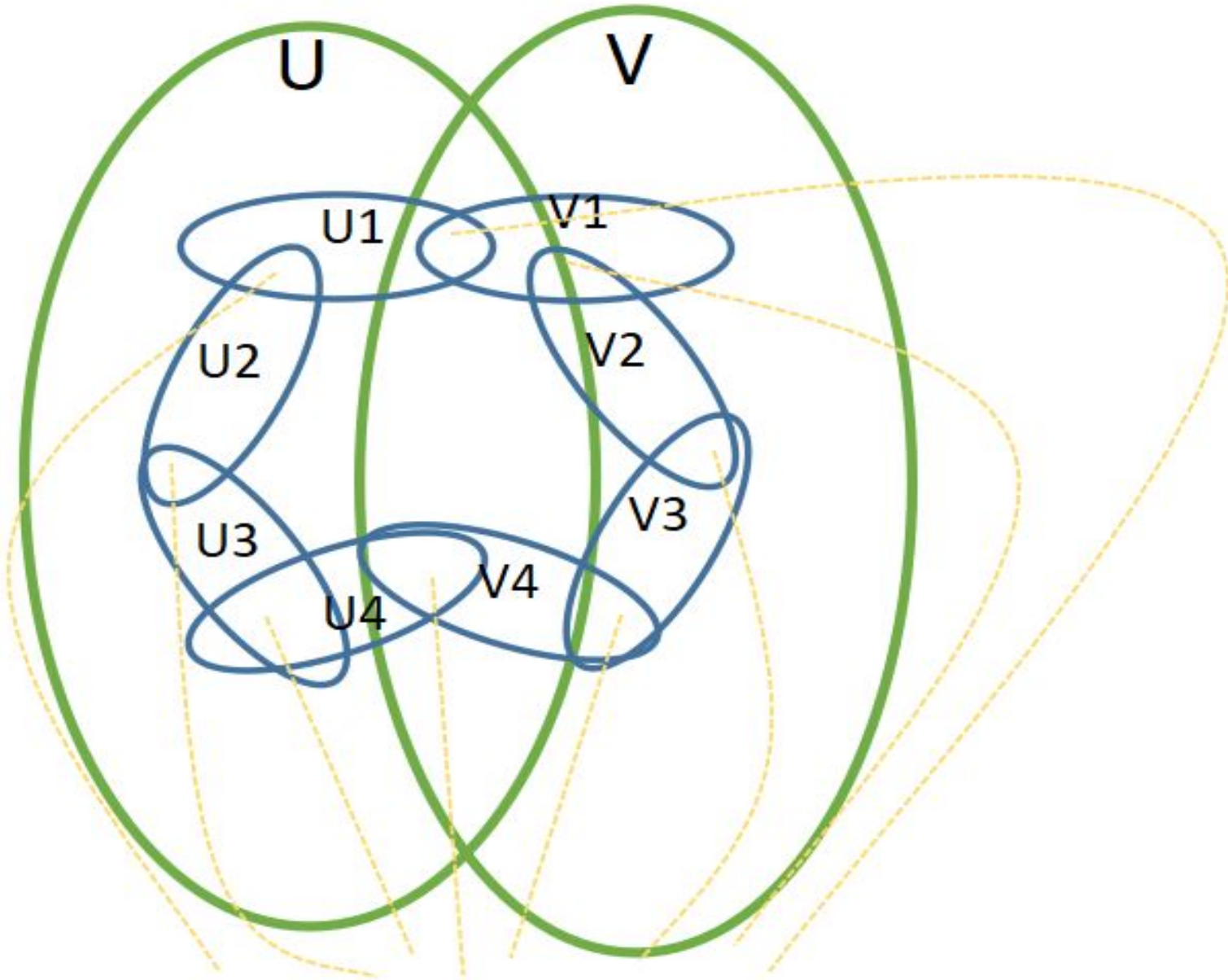
1. Map for X

When we consider a space X equipped with a $f : X \to Z$ to a parameter space Z, and we are given a map of coverings $U \to V$, there is a corresponding map of coverings $U \to V$ of the space X

The map of covering is given by requiring that the set $U_\alpha(i)$ is sent to the unique set of the form $V_{f(\beta)}(j)$ so that $U_\alpha(i) \subseteq V_{f(\beta)}(j)$, then

2. Basic Idea Here:
Use the covering and nerve of Z to study the nerve of X, and this idea derivate the statistical version of Mapper.

The main idea in passing from the topological version to the statistical version is that clustering should be regarded as the statistical version of the geometric notion of partitioning a space into its connected components.



All the overlap areas are interactions.

The cluster is applied to each subset of the covering(The covering will be get by the filter function which will be introduced later) and the clusters is called bins.

In general, the goal of a clustering algorithm is to partition observations into a certain number of clusters with similar attributes. It is, however, important to note that one can choose different notions of similarity and each of those choices can lead to different clustering outcome.

1. Single linkage

(1) Start with a matrix D of distances or similarities between N objects.

(2) Merge: Merge the two nearest objects, say C1 and C2 corresponding to the smallest entry $\epsilon > 0$ of the matrix D to form a new cluster [C1C2].

(3) Update distance matrix: Then the distances between the new cluster [C1C2] and say, C3 is computed by $d_{[C1C2]C3} = \min(d_{[C1C3]}, d_{[C2C3]})$. The matrix D is updated with the new distances

(4)Back to (1)

The result can be graphically represented in the form of a dendrogram.

2. Complete linkage

The difference with single linkage is $d_{[C1C2]C3} = \max(d_{[C1C3]}, d_{[C2C3]})$

3. Average linkage

The difference with single linkage is $d_{[C1C2]C3} = \frac{(d_{[C1C3]}, d_{[C2C3]})}{2}$

Many other clustering methods...

The desired characteristics of the clustering were:

1. Take the inter-point distance matrix ($D \in R^{N \times N}$) as an input. We did not want to be restricted to data in Euclidean Space.

2. Do not require specifying the number of clusters before hand.

Question: How can we do not require specifying the number of clusters before hand?

Answer:

1. The heuristic is that the inter-point distance within each cluster would be smaller than the distance between clusters, so shorter edges are required to connect points within each cluster, but relatively longer edges are required to merge the clusters.

2. If we look at the histogram of edge lengths in C, it is observed experimentally, that shorter edges which connect points within each cluster have a relatively smooth distribution and the edges which are required to merge the clusters are disjoint from this in the histogram.

Advantages of Single Linkage

Among the three clustering schemes mentioned, the single linkage is the continuous one. That is because it can be recovered in the following way: for a finite metric space X and $\epsilon > 0$, construct the Cech complex $C(X, \epsilon)$; the set of connected components of this complex is in bijection with the set of clusters of the single linkage algorithm for that ϵ. This means that the clusters of the single linkage algorithm correspond to components of a space.

Limitations of Single Linkage

(1) If the clusters have very different densities, it will tend to pick out clusters of high density only

(2) It is possible to construct examples where the clusters are distributed in such a way such that we recover the incorrect clustering

Due to such limitations, this part of the procedure is open to exploration and change in the future.

Recall that the in the theoritical version, first we map X to the parameter space Z and the covering of parameter space Z is mapped back to X. We need the map from X to Z...

Filter Functions

1. One of the basic idea of Mapper is convert X to a parameter space Z, and this process is done by Filter Function. Also note that, mapper is for homology, so coordinates are not preferred. Filter function should be constructed on distanc function of the metric space.

2. The choice of a filter functions is very important since a good function can help us to reveal some interesting geometrical information about the dataset.

1. Density Function

For $\epsilon > 0$ consider estimating density using a Gaussian kernel as:

$$\mathrm{f}_{\varepsilon}(\mathrm{x}) = \mathrm{C}_{\varepsilon} \sum_{\mathrm{y}} \exp\left(\frac{-\mathrm{d}(\mathrm{x},\mathrm{y})^2}{\varepsilon}\right) \tag{13}$$

where $\mathrm{x}, \mathrm{y} \in \mathrm{X}$ and C_{ϵ} is a constant such that $\int \mathrm{f}_{\epsilon}(\mathrm{x})\mathrm{dx} = 1$. In this formulation ϵ controls the smoothness of the estimate of the density function on the data set.

2. Eccentricity Function The basic idea is to identify points which are, in an intuitive sense, far from the center, without actually identifying an actual center point. Given p with $1 \leq p < +\infty$, we set:

$$E_p(x) = \left(\frac{\sum_{y \in X} d(x,y)^p}{N} \right)^{\frac{1}{p}} \tag{14}$$

where x, y X. We may extend the definition to $p = +\infty$ by setting $E_\infty(x) = \max_{x \in X} d(x, x)$. In the case of a Gaussian distribution, this function is clearly negatively correlated with density. In general, it tends to take larger values on points which are far removed from a "center" .

Recall that when we mention the example of hand at the beginning, but it depends on the x coordinate. We have some substites here.

3. Filter Function Based on SVD

(1) Given a matrix of data points one can apply singular value decomposition in order to obtain the k-th eigenvector of a distance matrix, for example the principal eigenvector corresponds to the largest eigenvalue in magnitude.

(2) this projection can serve as a filter function and we can therefore produce a topological summary. Another projection yields a different filter function and therefore possibly a different-looking topological summary compared to the previous one

(3) Step:

A. Calculate distance matrix $[d(x_i, x_j)]_{ij}$ (15)

B. SVD and find a principle vector v_k (16)

C. Define the filter function as $f(x_i) = v_k \cdot (..., d(x_i, x_j), ...)$ (17)

1. Graph Laplacians

Graph Laplacians is also based on SVD. The vertex set of this graph is the set of all points in the point cloud data X, and the weight of the edge between points x, y X is:

$$\mathrm{w}(\mathrm{x},\mathrm{y}) = \mathrm{k}(\mathrm{d}(\mathrm{x},\mathrm{y})) \tag{18}$$

where d denotes the distance function in the point cloud data and k is, roughly, a "smoothing kernel" such as a Gaussian kernel. A (normalized) graph Laplacian matrix is computed as:

$$\mathrm{L}(\mathrm{x},\mathrm{y}) = \frac{\mathrm{w}(\mathrm{x},\mathrm{y})}{\sqrt{\sum_{\mathrm{z}} \mathrm{w}(\mathrm{x},\mathrm{z})}\sqrt{\sum_{\mathrm{z}} \mathrm{w}(\mathrm{y},\mathrm{z})}} \tag{19}$$

Now, the eigenvectors of the normalized graph Laplacian matrix gives us a set of orthogonal vectors which encode interesting geometric information, and can be used as filter functions on the data.

1. Calculate a filter function of each point

2. Separate points into N bins based on the value of the filter function, the bins should have p overlap with its adjacent interval

3. Select a clustering scheme and apply it to each bin

4. Draw a graph with the clustering results of 3 as the vertex. If two clusters(even in different bins) have not-empty adjacent, connect the two vertexes.

Example of 1-Dimensional Mapper

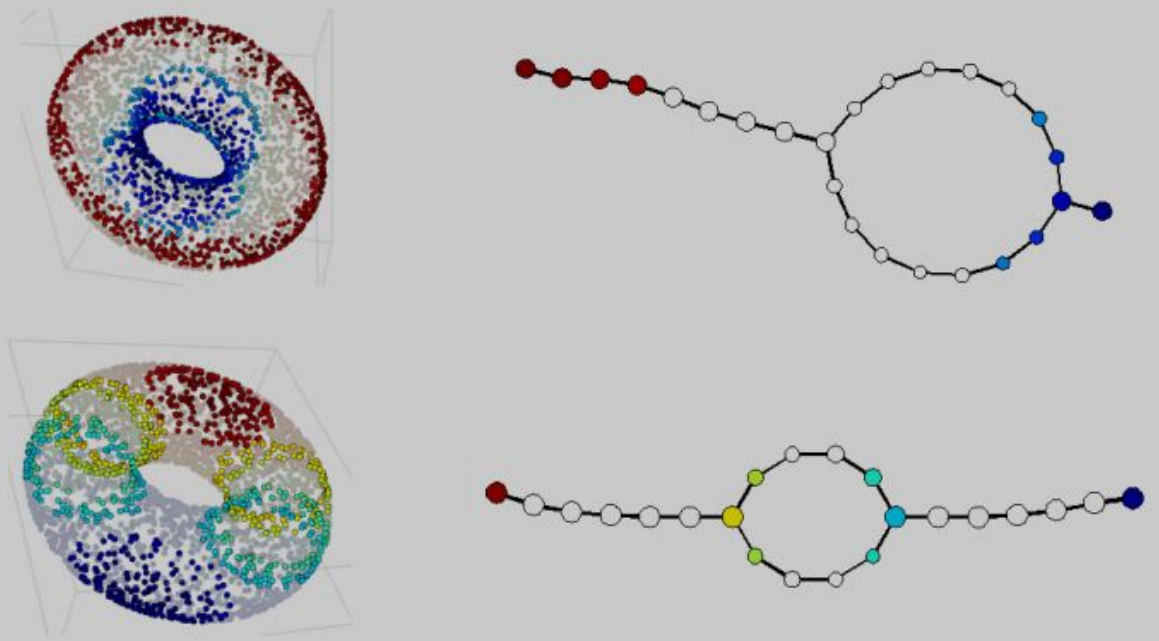


FIGURE 4.8: Topological summary of a sample of torus colored by first SVD filter function (top) and second SVD (bottom).

Example of 1-Dimensional Mapper

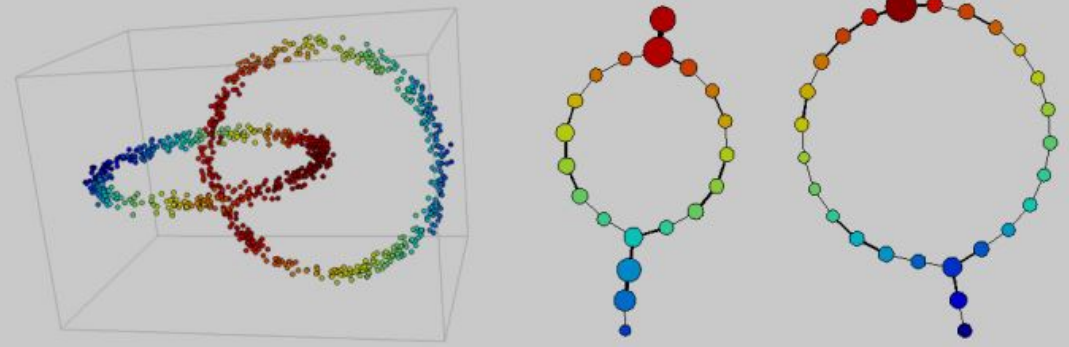


FIGURE 4.9: Topological summary of a sample of the chain link data colored by the SVD filter function.

Example of 1-Dimensional Mapper

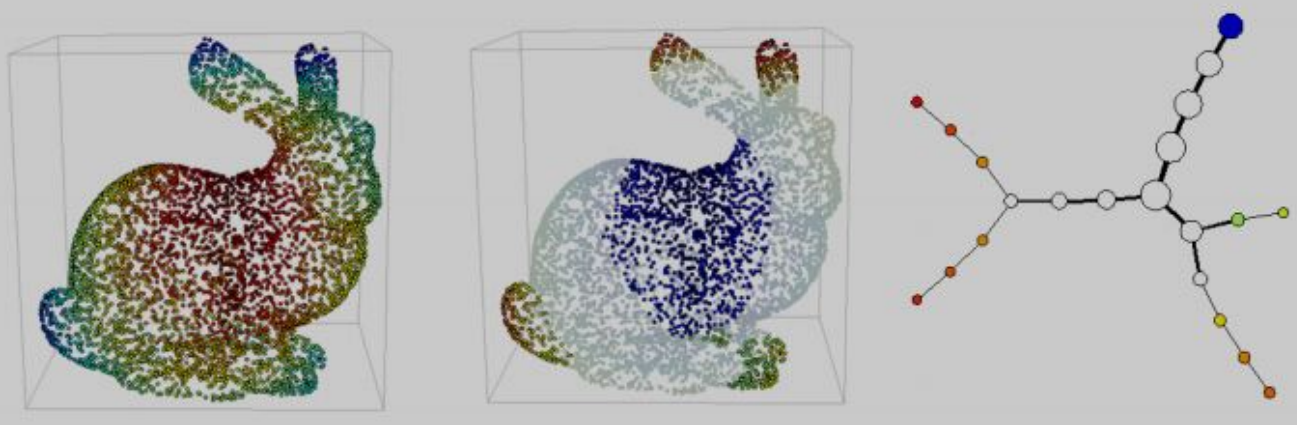


FIGURE 4.11: Topological summary of a sample of 5000 randomly chosen points of the original Stanford bunny point cloud coloured by the first SVD filter function.

1. What's the dimension of a mapper?
Ans: How many filter are used. Which determine the dimension of the parameter space Z

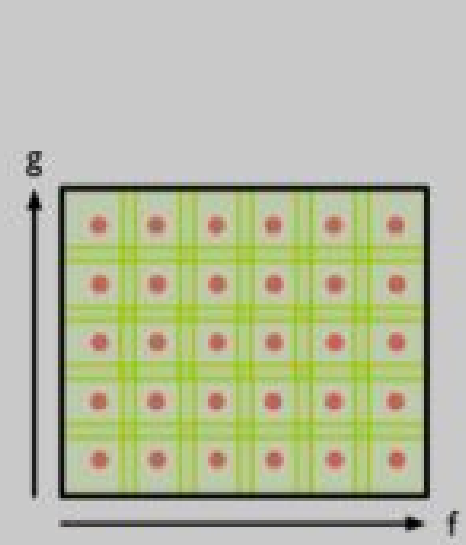
2. What's the limitation of 1-Dimensional mapper?

Ans:
(1) Using a single function as a filter we get as output a complex in which the highest dimension of simplices is 1 (edges in a graph).

(2) Qualitatively, the only information we get out of this is the number of components, the number of loops and knowledge about structure of the component flares etc.

1. To get information about higher dimensional voids in the data one would need to build a higher dimensional complex using more functions on the data

2. One natural way of building higher dimensional complexes is to associate many functions with each data point instead of just one. If we used M functions and let R^M to be our parameter space, then we would have to find a covering of an M dimensional hypercube which is defined by the ranges of the M functions.



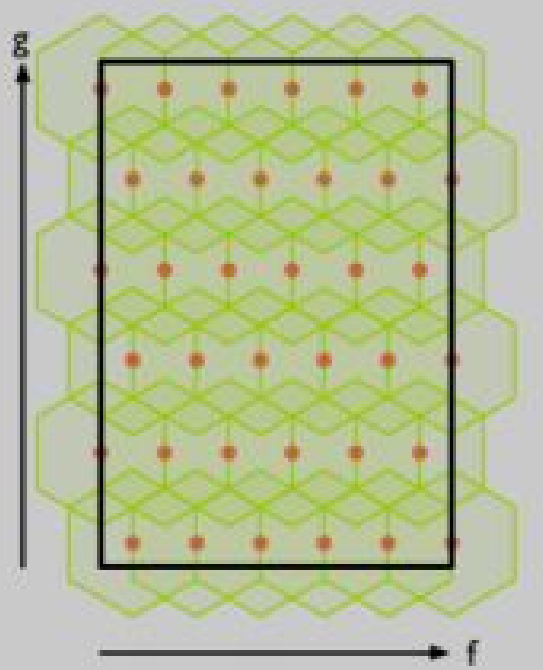


Figure 2: *Covering the range of two functions f and g. The area which needs to be covered is $[\min f, \max f] \times [\min g, \max g]$. On the left is a covering using rectangles and on the right is a covering using hexagons. The constraints on*

We now describe the Mapper algorithm using two functions and the parameter space R^2, Consider two functions on each data point, and the range of these being covered by rectangles. Define a region $R = [minf_1, maxf_1] \times [minf_2, maxf_2]$. Now say we have a covering $\cup_{i,j} A_{ij}$ such that each $A_{i,j}, A_{i+1,j}$ intersect and each $A_{i,j}, A_{i,j+1}$ intersect. An algorithm for building a reduced simplicial complex is:

1. Cluster: calculate the cluster for each A_{ij}
2. Add 1-simplex

Consider sets $\{A_{i,j}, A_{i+1,j}, A_{i,j+1}, A_{i+1,j+1}\}$. If any 2 clusters in these sets intersect, add a 1-simplex.

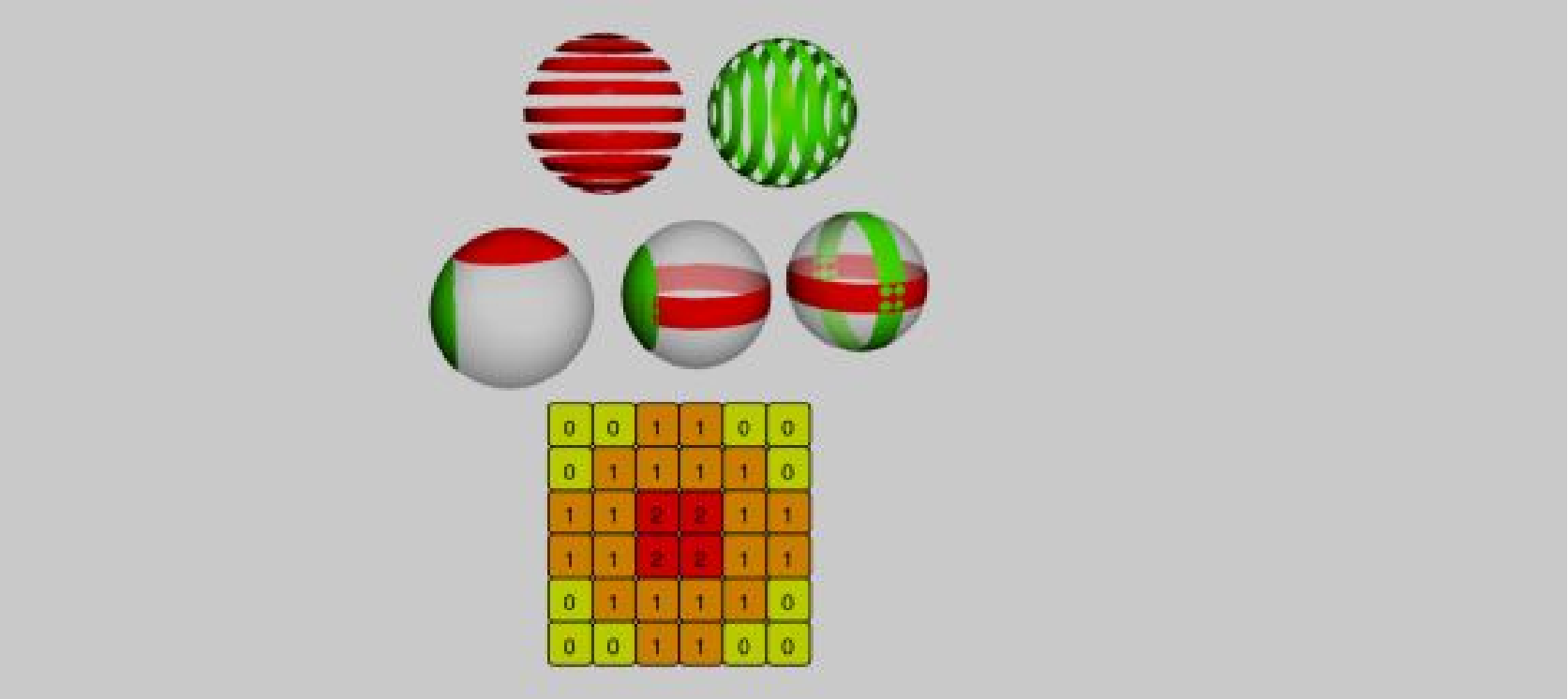
3. Add 2-simplex

Consider sets $\{A_{i,j}, A_{i+1,j}, A_{i,j+1}, A_{i+1,j+1}\}$. If any 3 clusters in these sets intersect, add a 2-simplex.

4. Add 3-simplex

Consider sets $\{A_{i,j}, A_{i+1,j}, A_{i,j+1}, A_{i+1,j+1}\}$. If any 4 clusters in these sets intersect, add a 3-simplex.

Example of Higher-Dimensional Mapper



1. The functions are $f_1(x) = x_3$ and $f_2(x) = x_1$, where $x = (x_1, x_2, x_3)$
2. Consider the intervals of the rage of f_1 and f_2, clearly only three possibilities exist:

(1) The intersection is empty, and we get no clusters.
(2) The intersection contains only one cluster.
(3) The intersection contains two clusters.