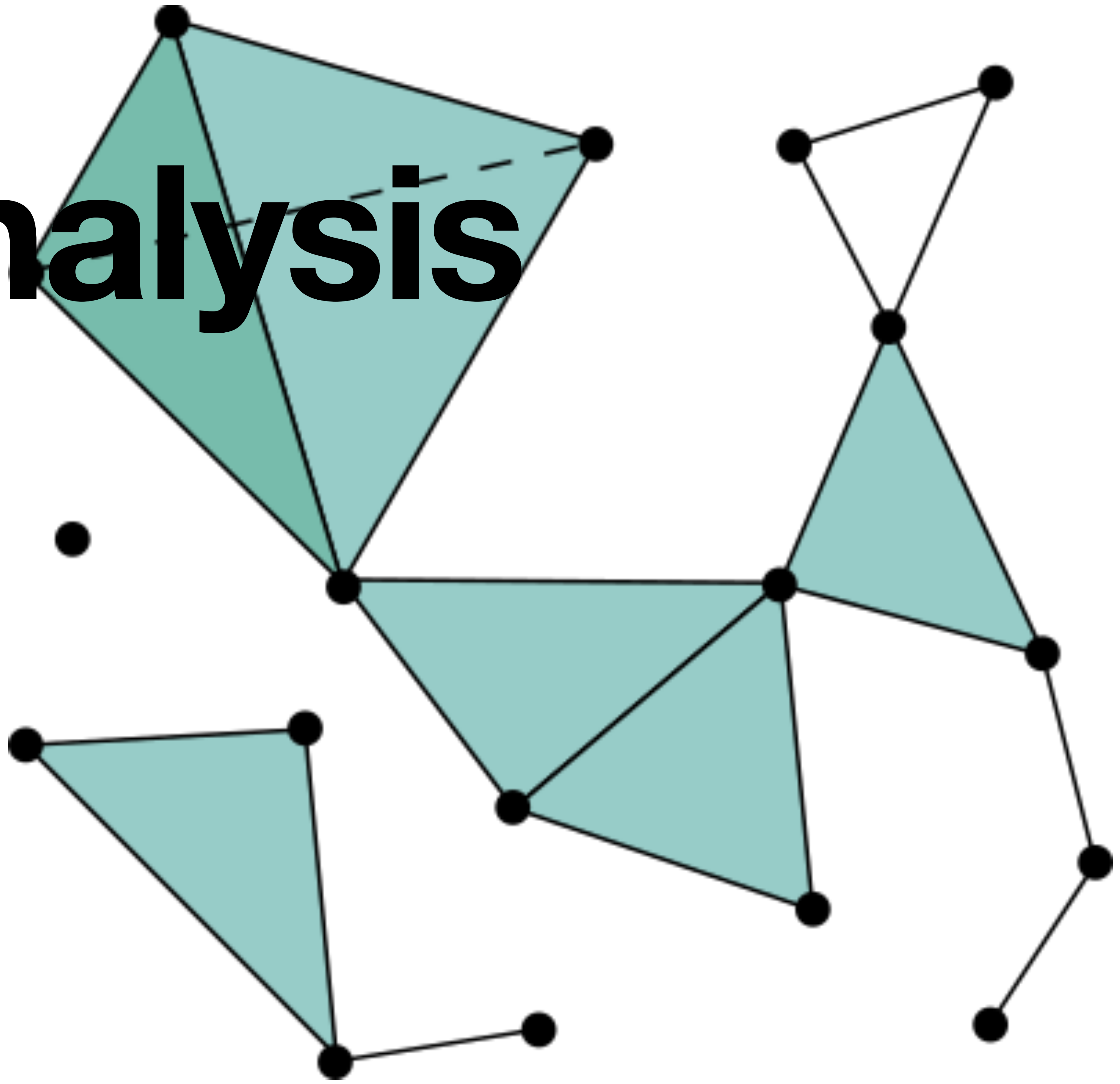
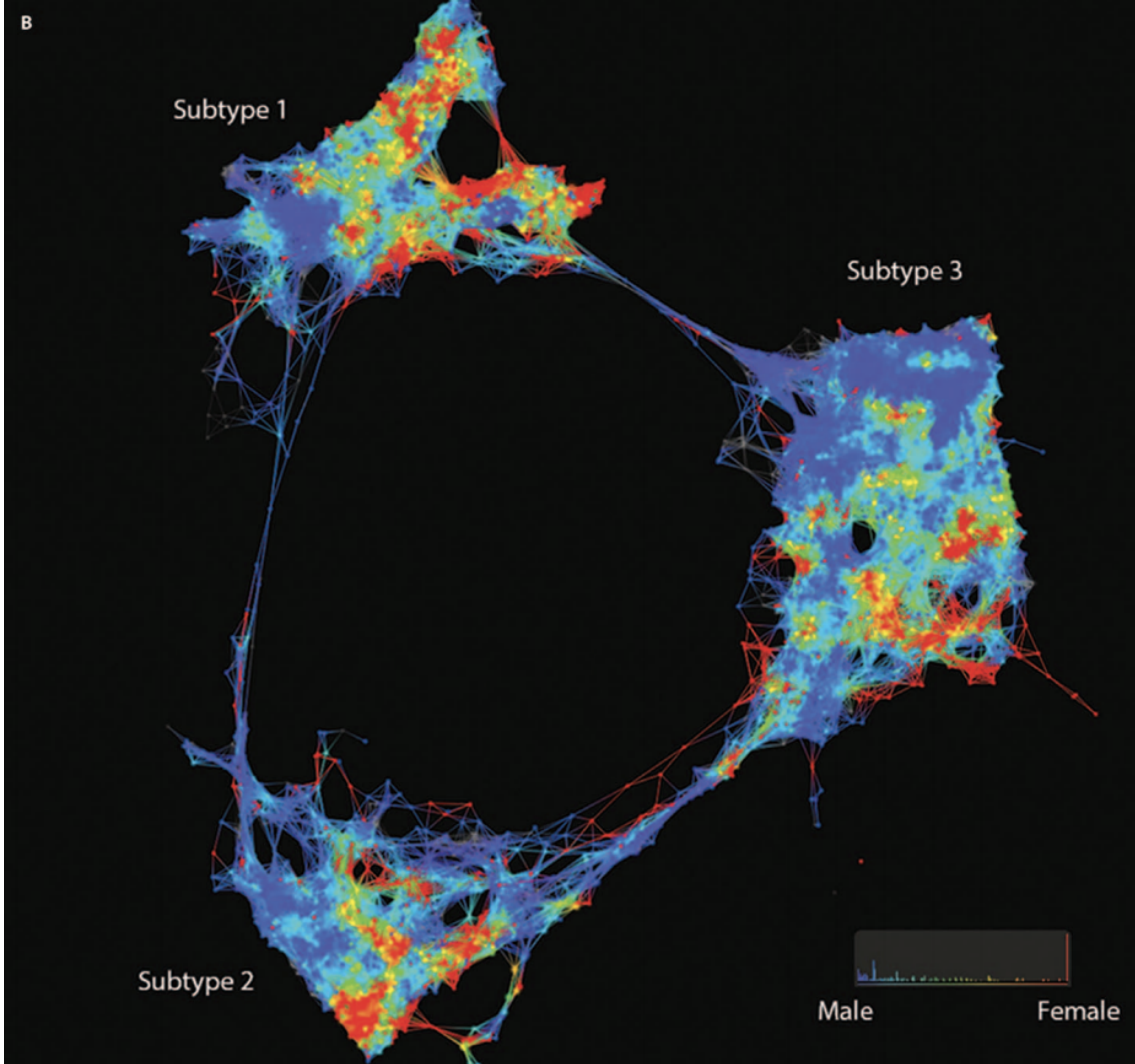
Topological Data Analysis



Pol Llopart Mirambell

Index

- What is topology
- Topology of data
- Persistent homology
- Mapper algorithm
- Interesting links



What is topology

Topology

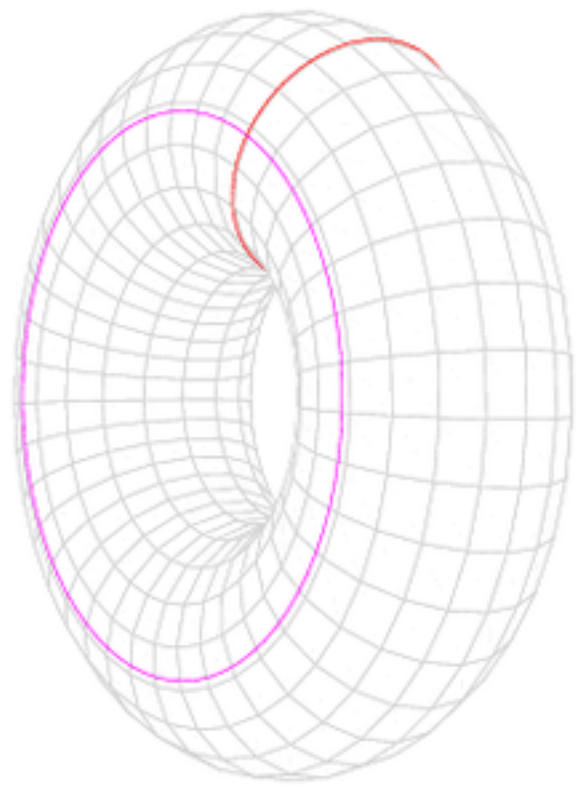
Topology started with Leonhard Euler and the famous problem of the Seven Bridges of Königsberg: Can one construct a path that crosses each bridge exactly once and reaches all islands?

In mathematics, topology (from the Greek words τόπος, 'place', and λόγος, 'study') is concerned with the properties of a geometric object that are preserved under continuous deformations, such as stretching, twisting, crumpling and bending, but not tearing or gluing.

Homology

In mathematics, homology is a general way of associating a sequence of algebraic objects such as abelian groups or modules to other mathematical objects such as topological spaces.

In algebraic topology, the Betti numbers are used to distinguish topological spaces based on the connectivity of n-dimensional simplicial complexes. For the most reasonable finite-dimensional spaces (such as compact manifolds, finite simplicial complexes or CW complexes), the sequence of Betti numbers is 0 from some point onward (Betti numbers vanish above the dimension of a space), and they are all finite.



For a torus, the first Betti number is b1 = 2 , which can be intuitively thought of as the number of circular "holes"

Homology

Informally, the kth Betti number refers to the number of k-dimensional holes on a topological surface. The first few Betti numbers have the following definitions for 0-dimensional, 1-dimensional, and 2-dimensional simplicial complexes:

b0 is the number of connected components

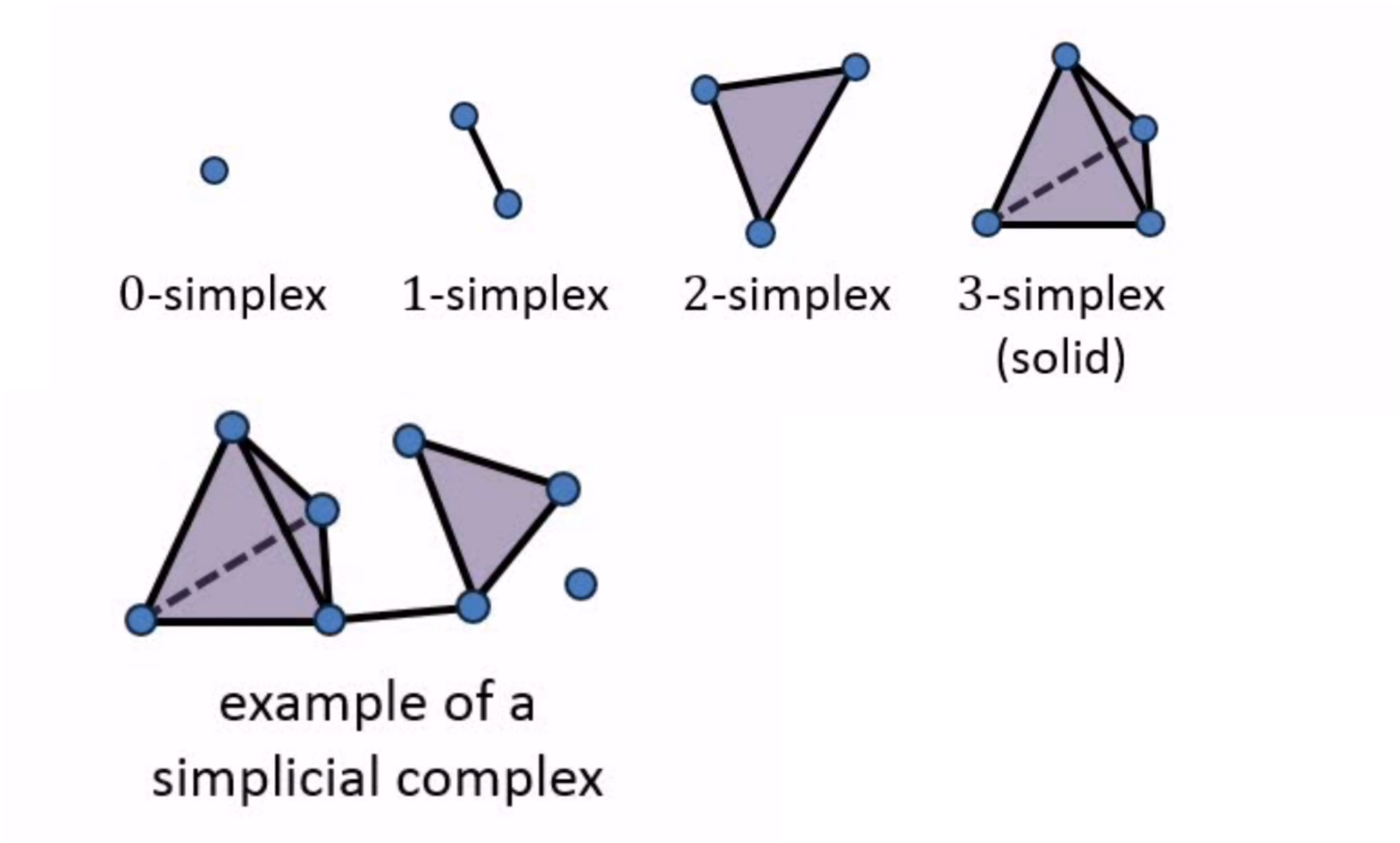
b1 is the number of one-dimensional or "circular" holes

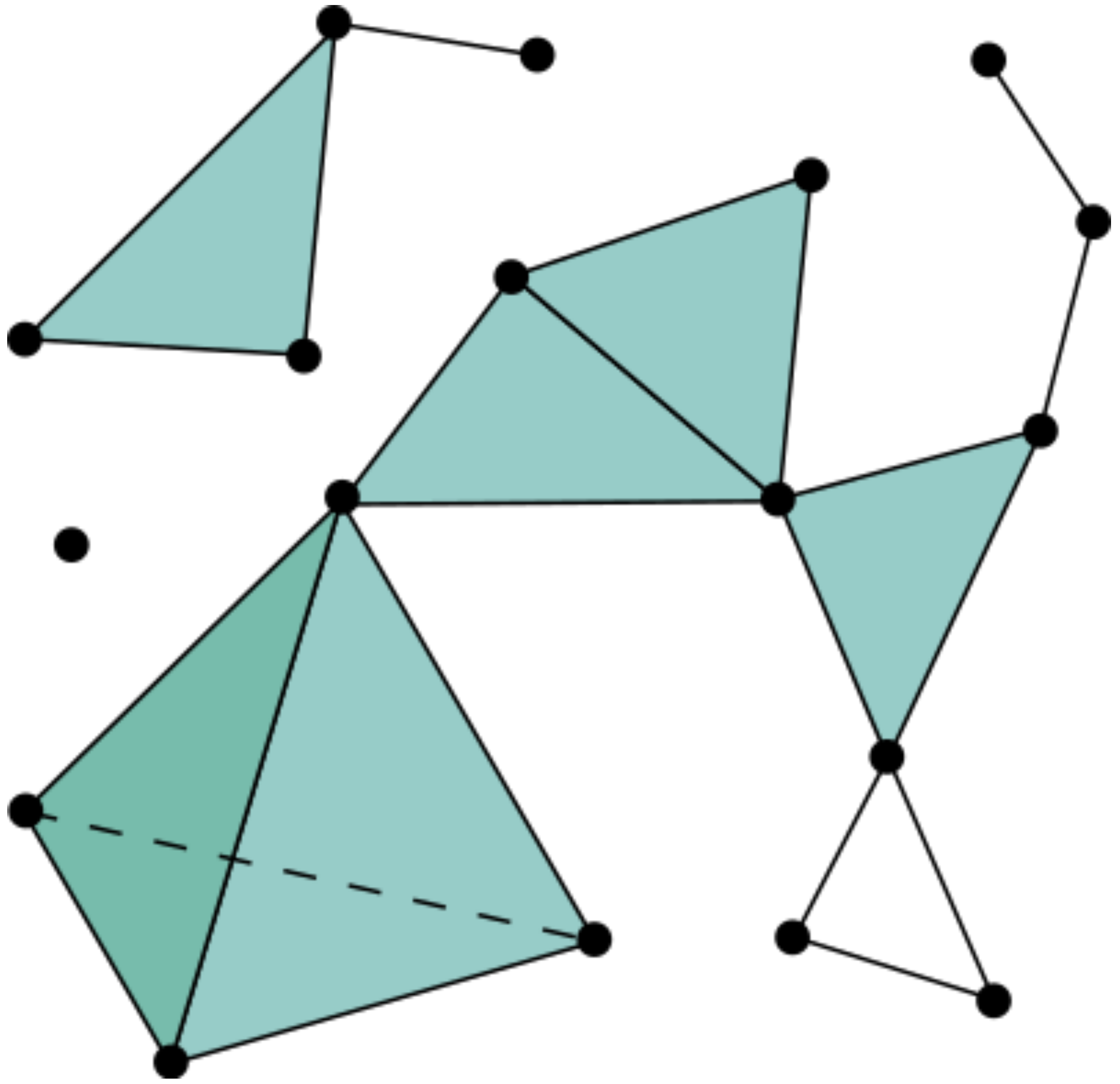
b2 is the number of two-dimensional "voids" or "cavities"

Thus, for example, a torus has one connected surface component so b0 = 1, two "circular" holes (one equatorial and one meridional) so b1 = 2, and a single cavity enclosed within the surface so b2 = 1

Simplical complex

In mathematics, a simplicial complex is a set composed of points, line segments, triangles, and their n-dimensional counterparts

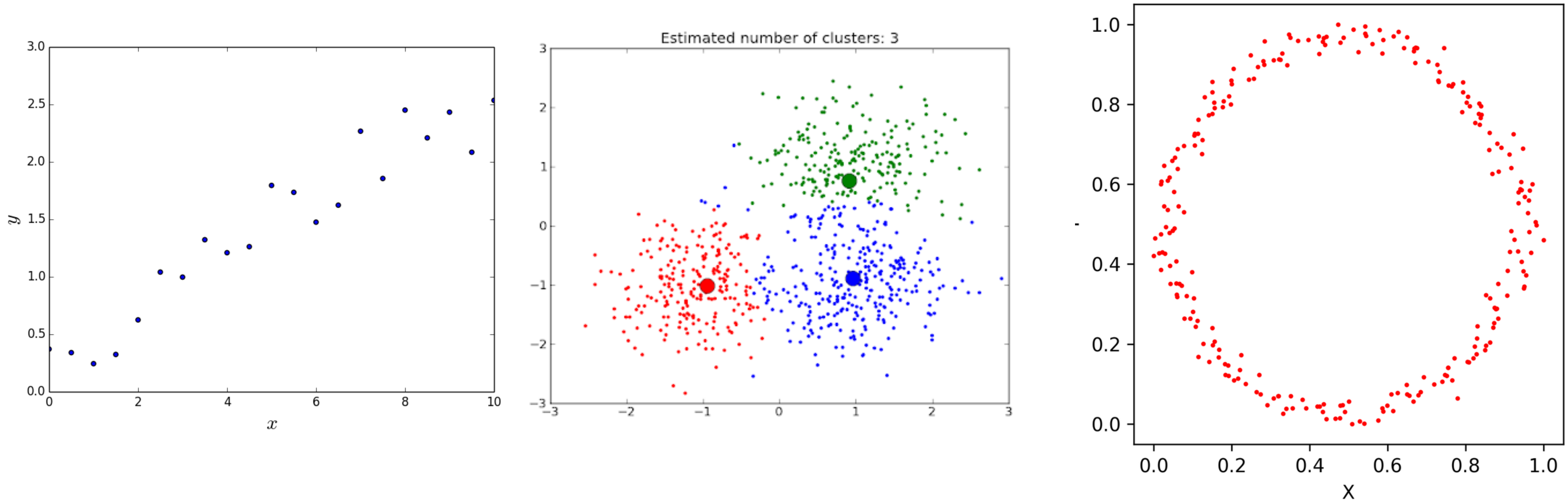




Topology of data

Learning from data?

Data has a shape, by understanding its shape we can get useful information from it



Learning from data?

Fundamental shapes of data that can be studied with topology:

- linearities
- non-linearities
- clusters
- flares
- loops

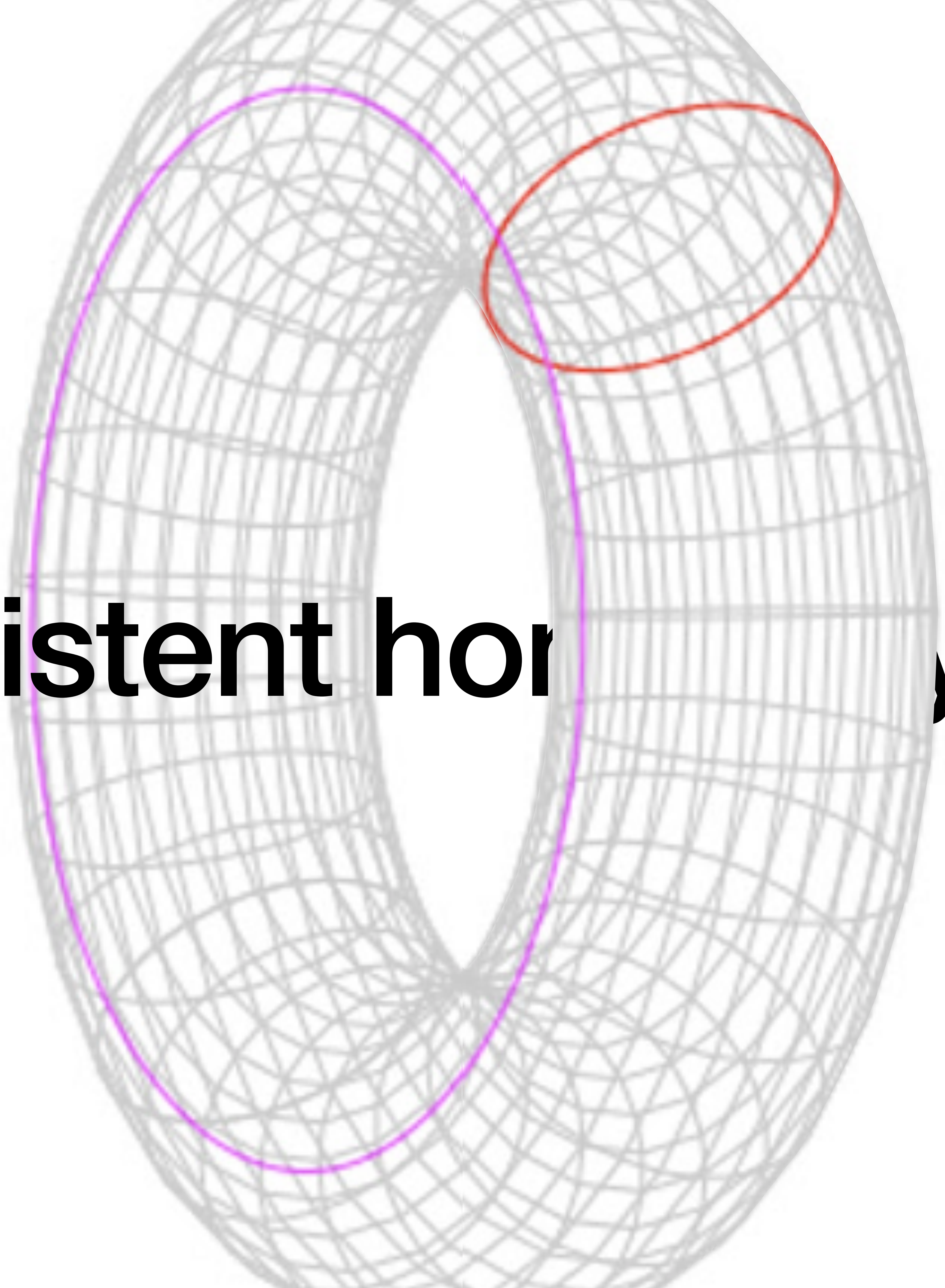
Real-world data is often complex and contains multiple different fundamental shapes.

Topological Data Analysis gives us a set of tools to understand te shape and connectivity of our data

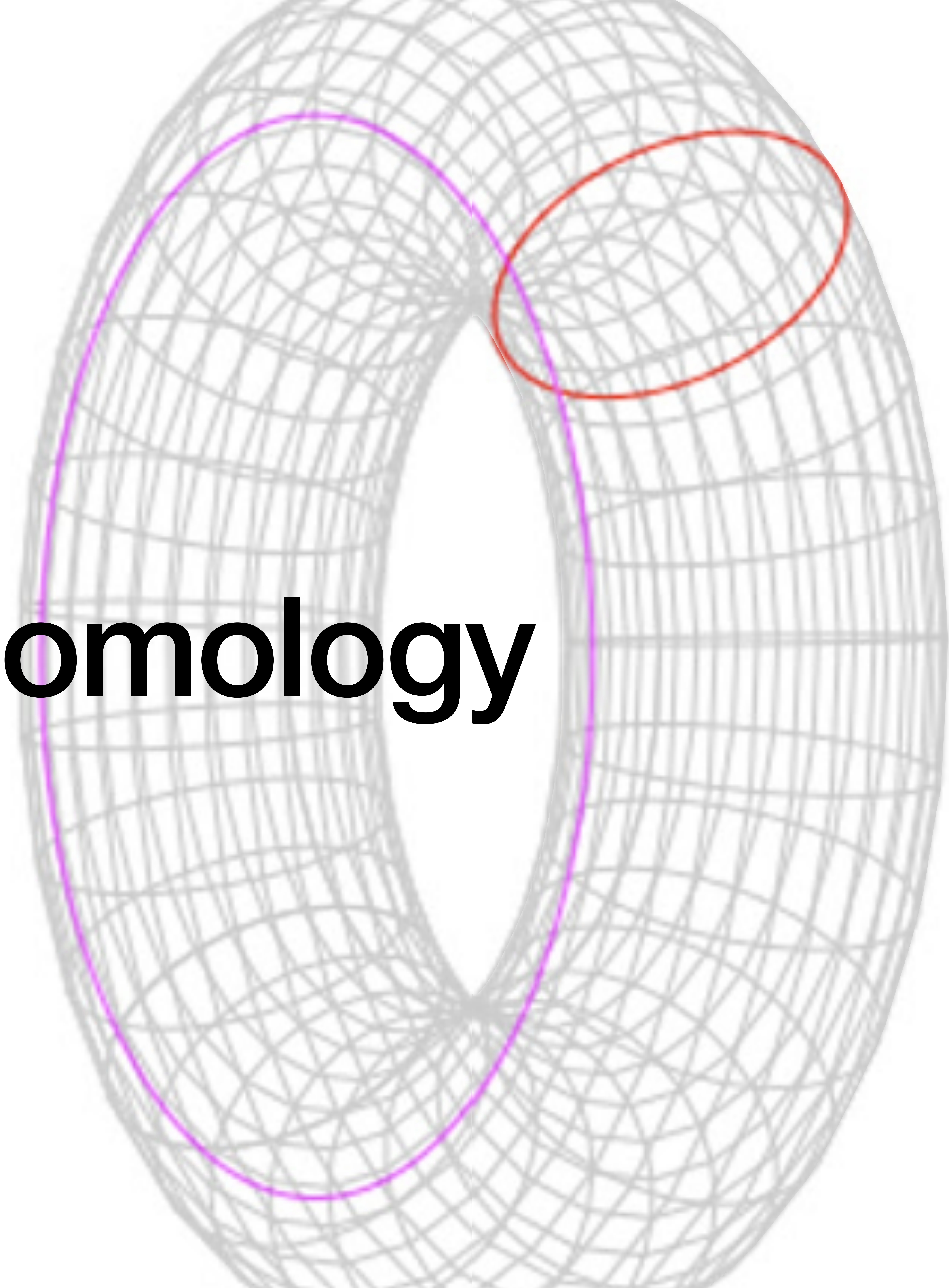
- Persistent homology

- Mapper algorithm

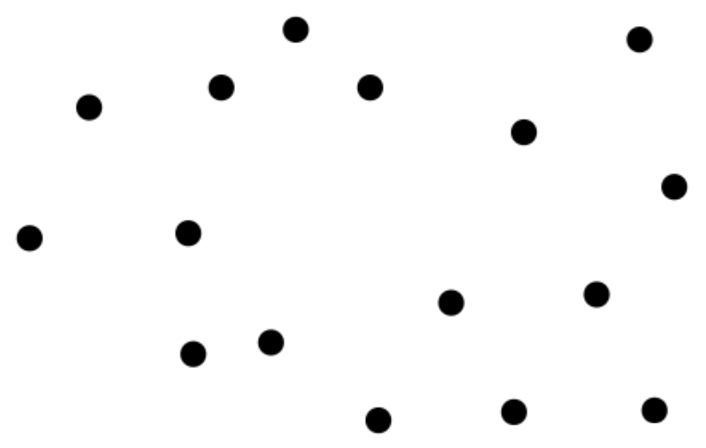
Persistent hor y



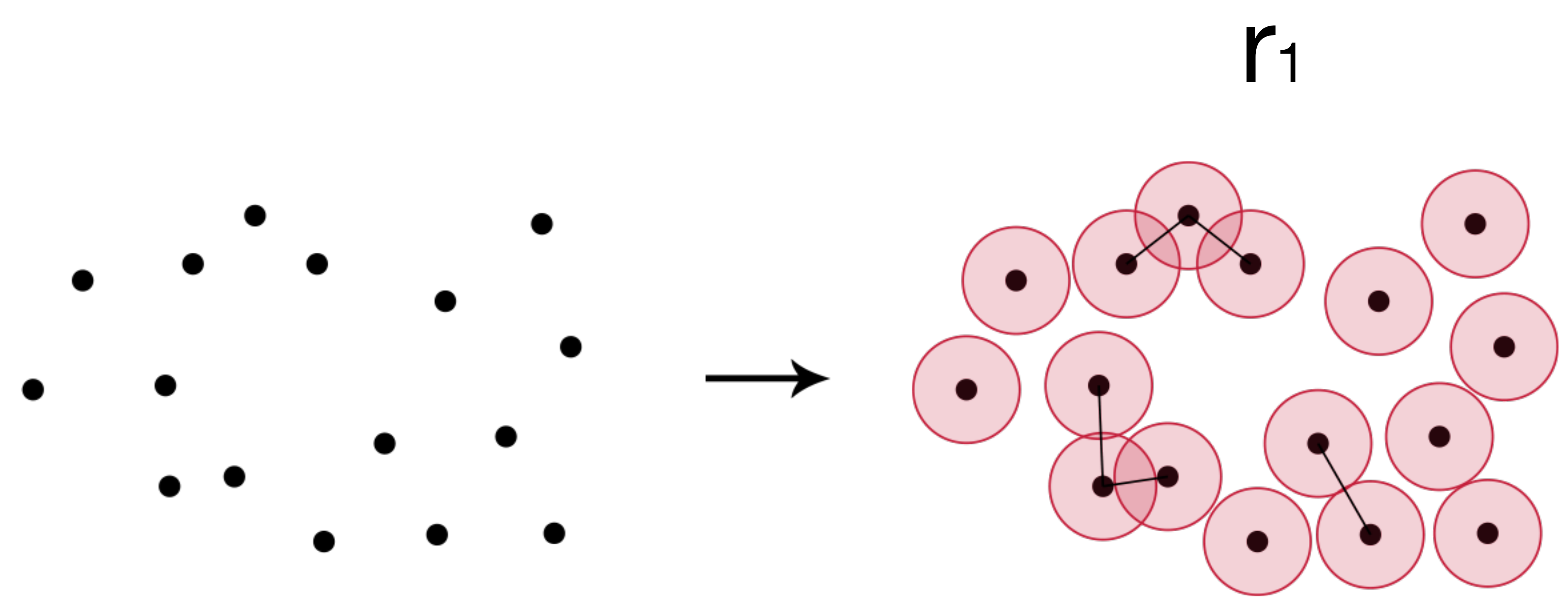
Persistent homology



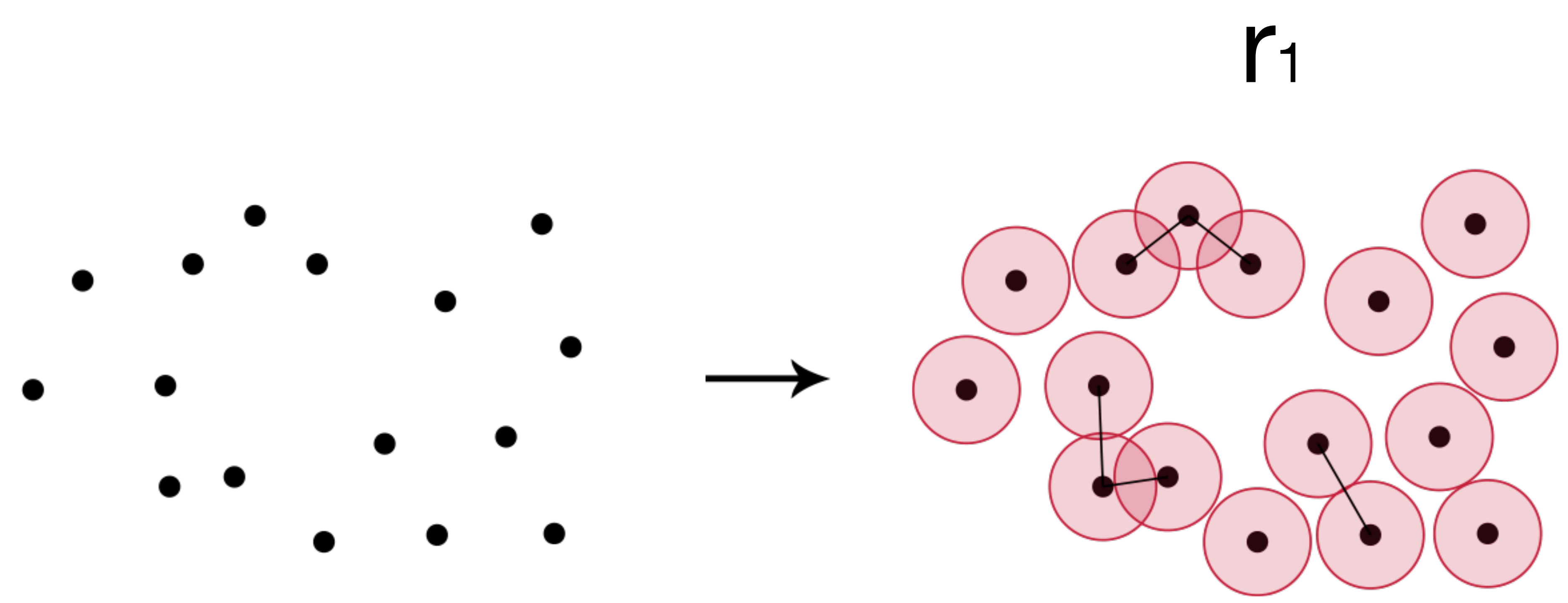
Persistent homology



Persistent homology

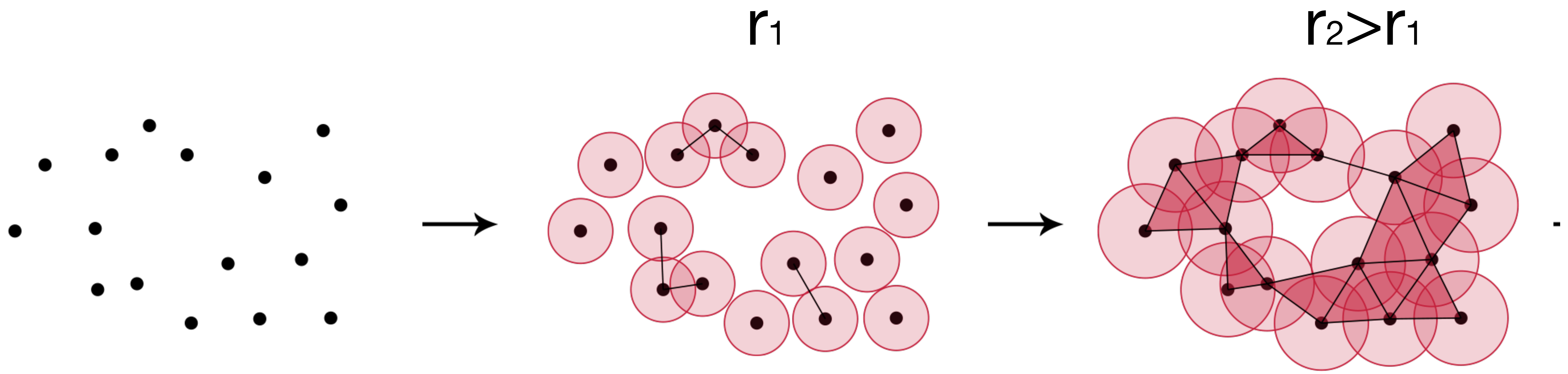


Persistent homology

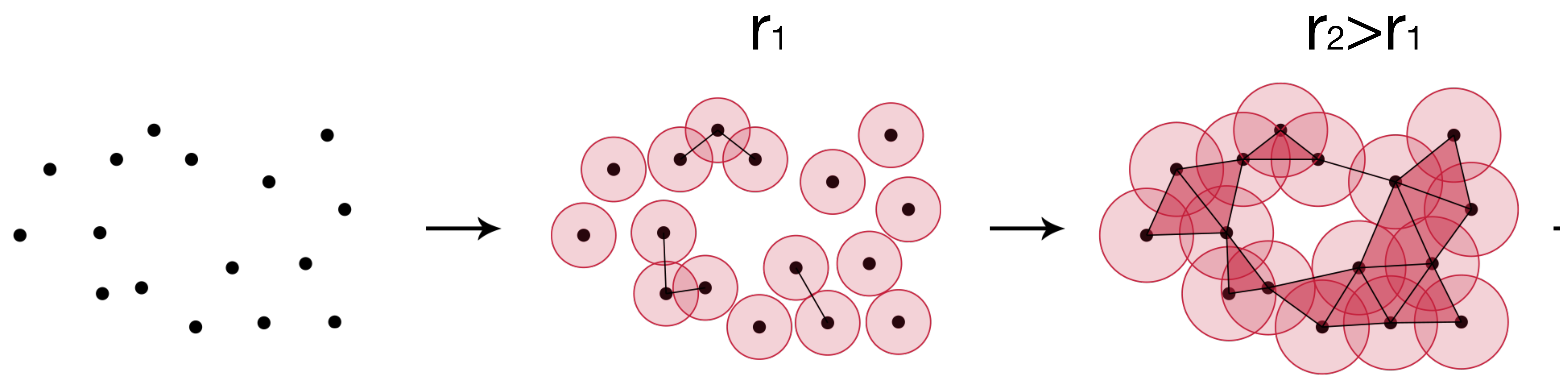


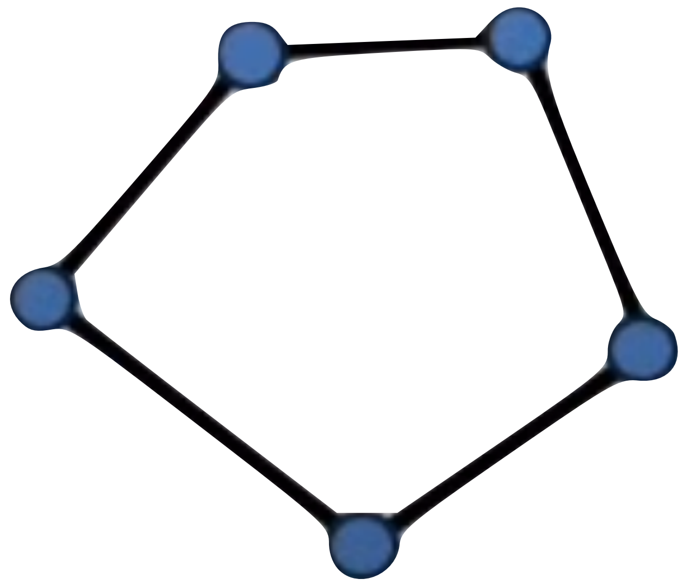
0d persistent homology in Euclidean space can best be explained as growing balls simultaneously around each point. The key focus of 0d persistent homology here is connected components— as the balls around the points expand, 0d persistent homology notes when the balls touch.

Persistent homology

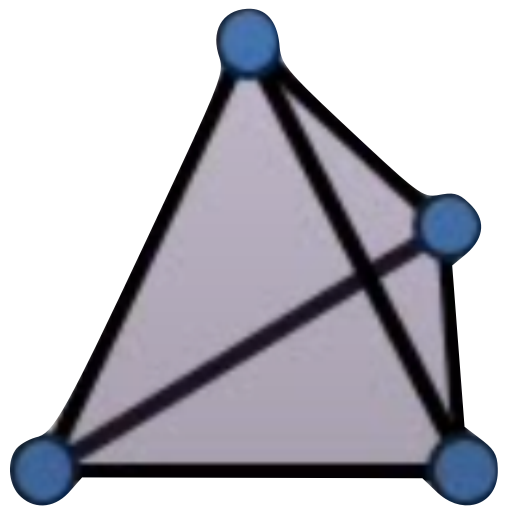


Persistent homology





Hole H_1



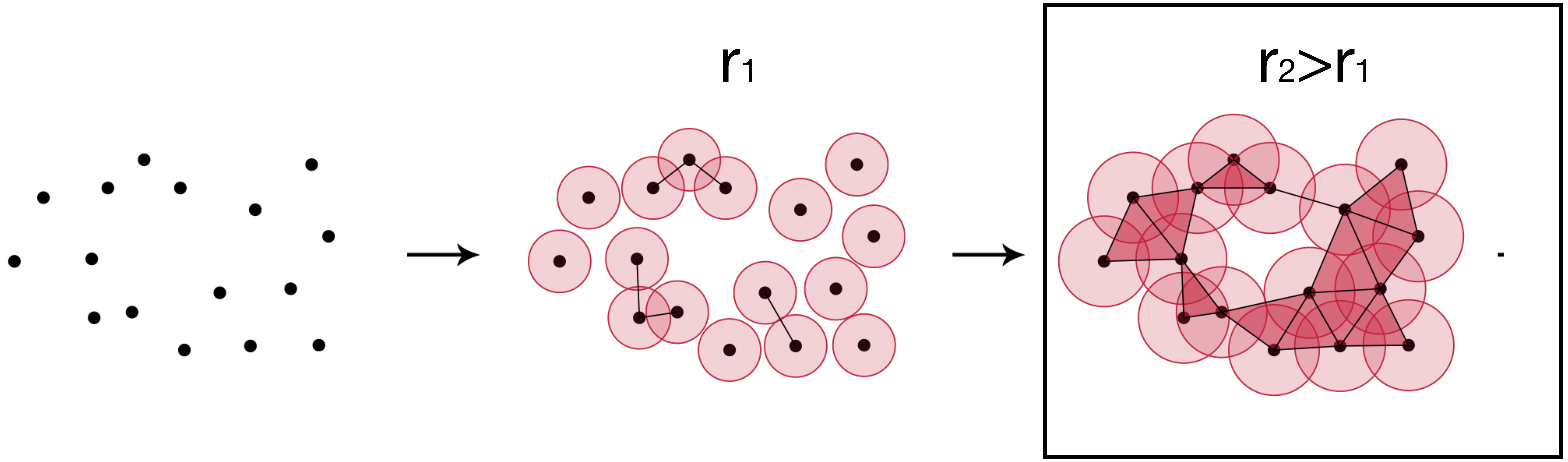
Void H_2

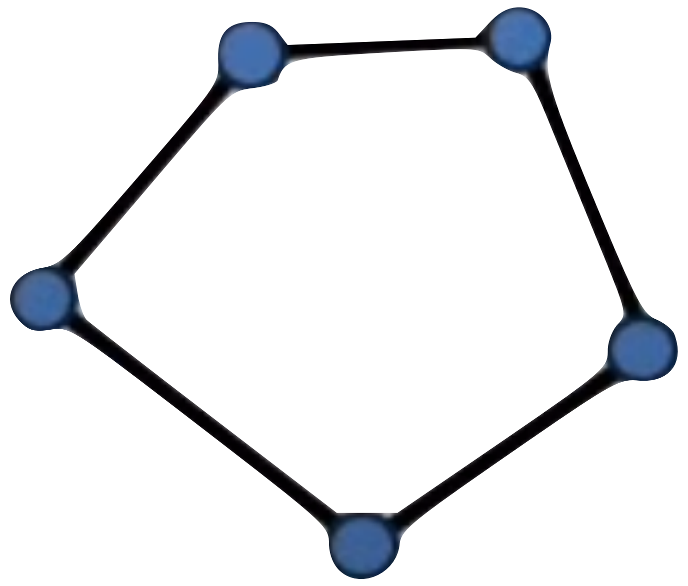
Homology tells us:

$H_0 = 1$

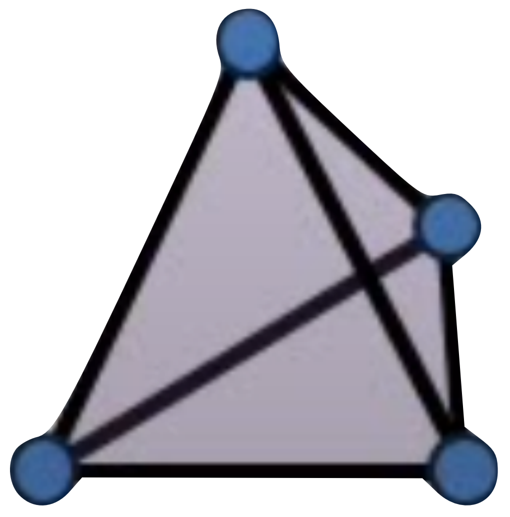
$H_1 = 1$

Persistent homology





Hole H_1



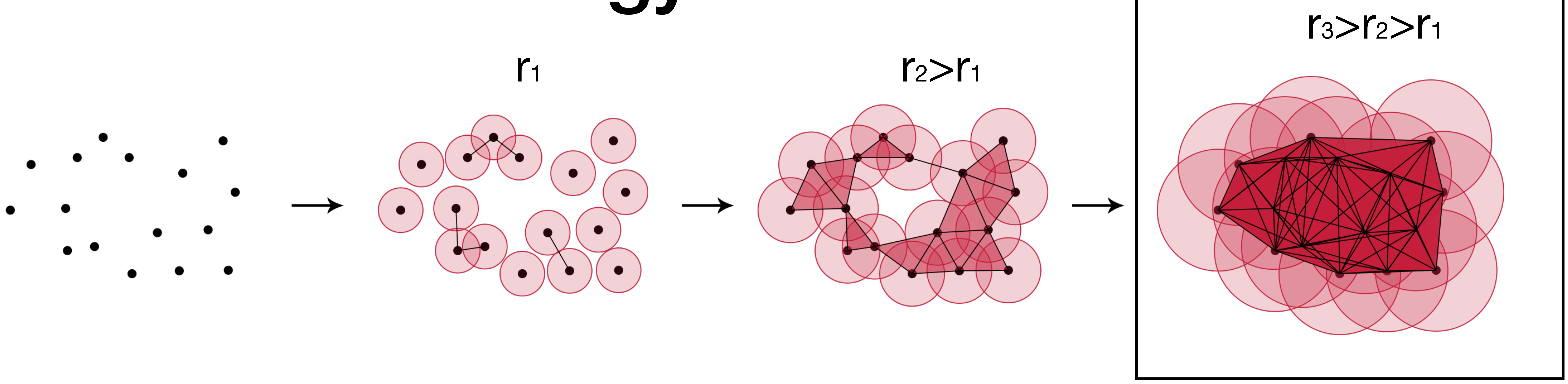
Void H_2

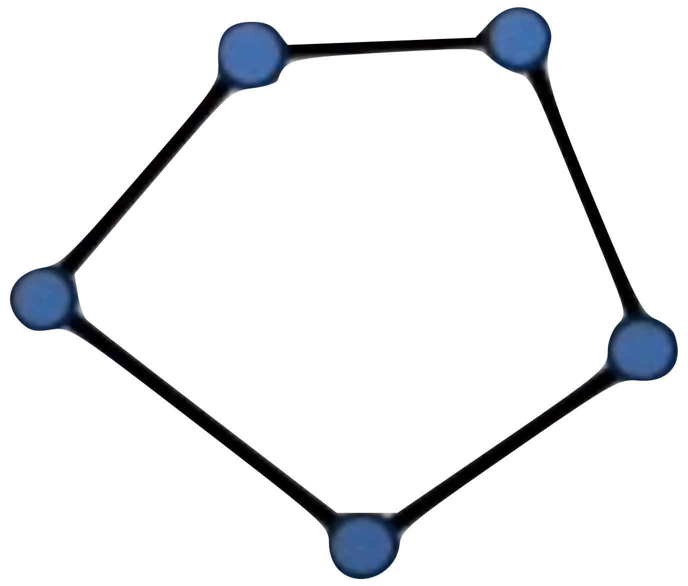
Homology tells us:

$H_0 = 1$

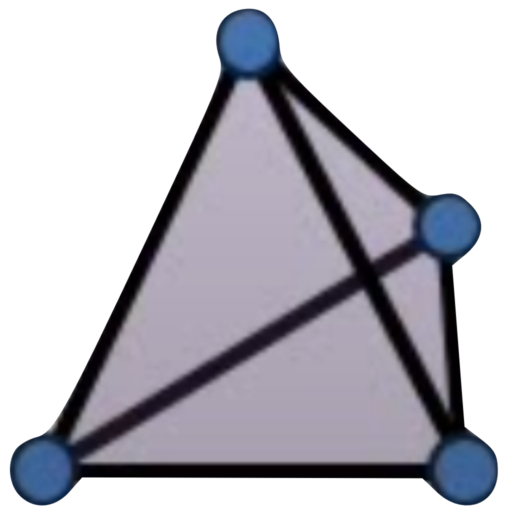
$H_1 = 1$

Persistent homology





Hole H_1



Void H_2

Homology tells us:

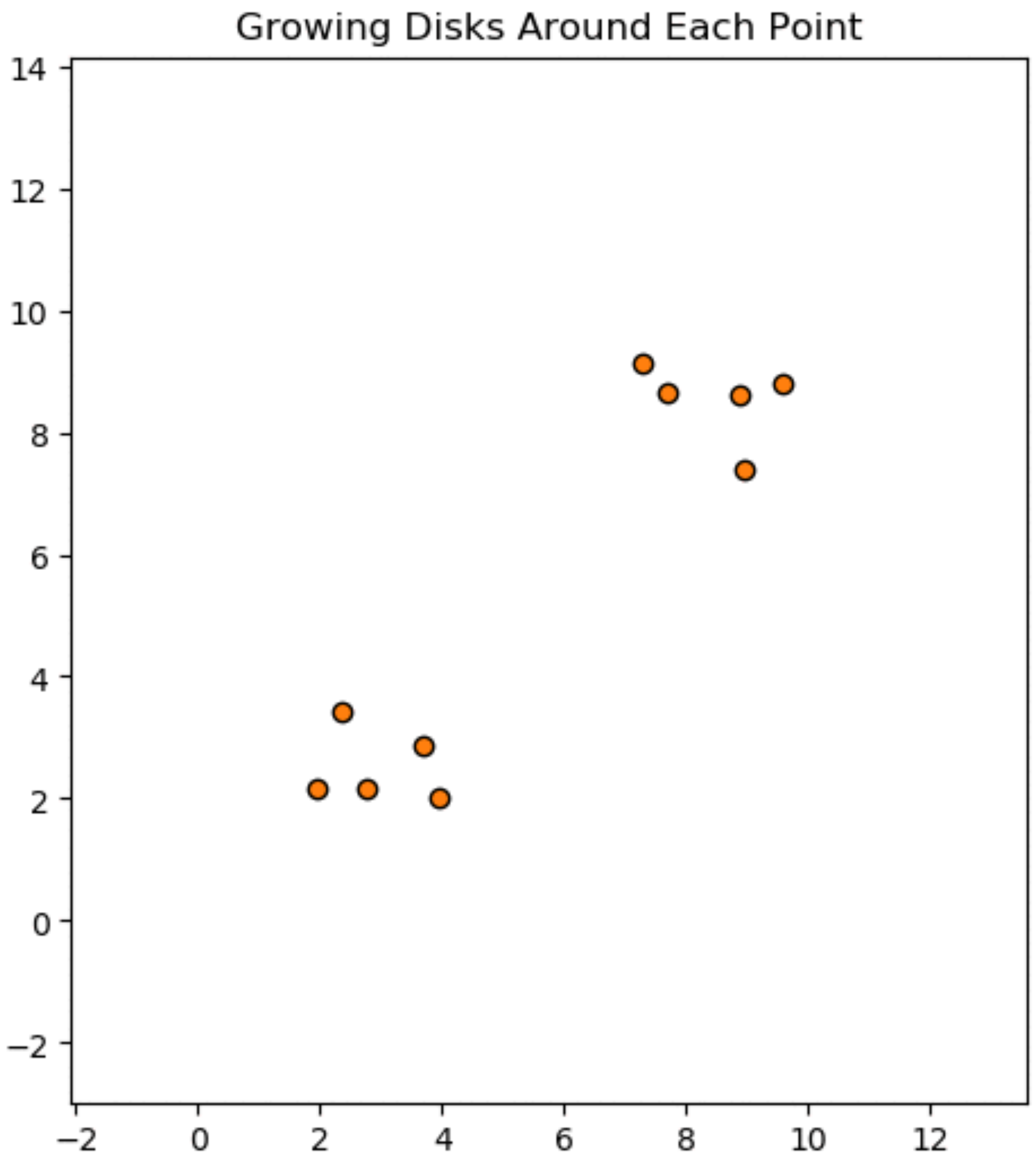
$H_0 = 0$

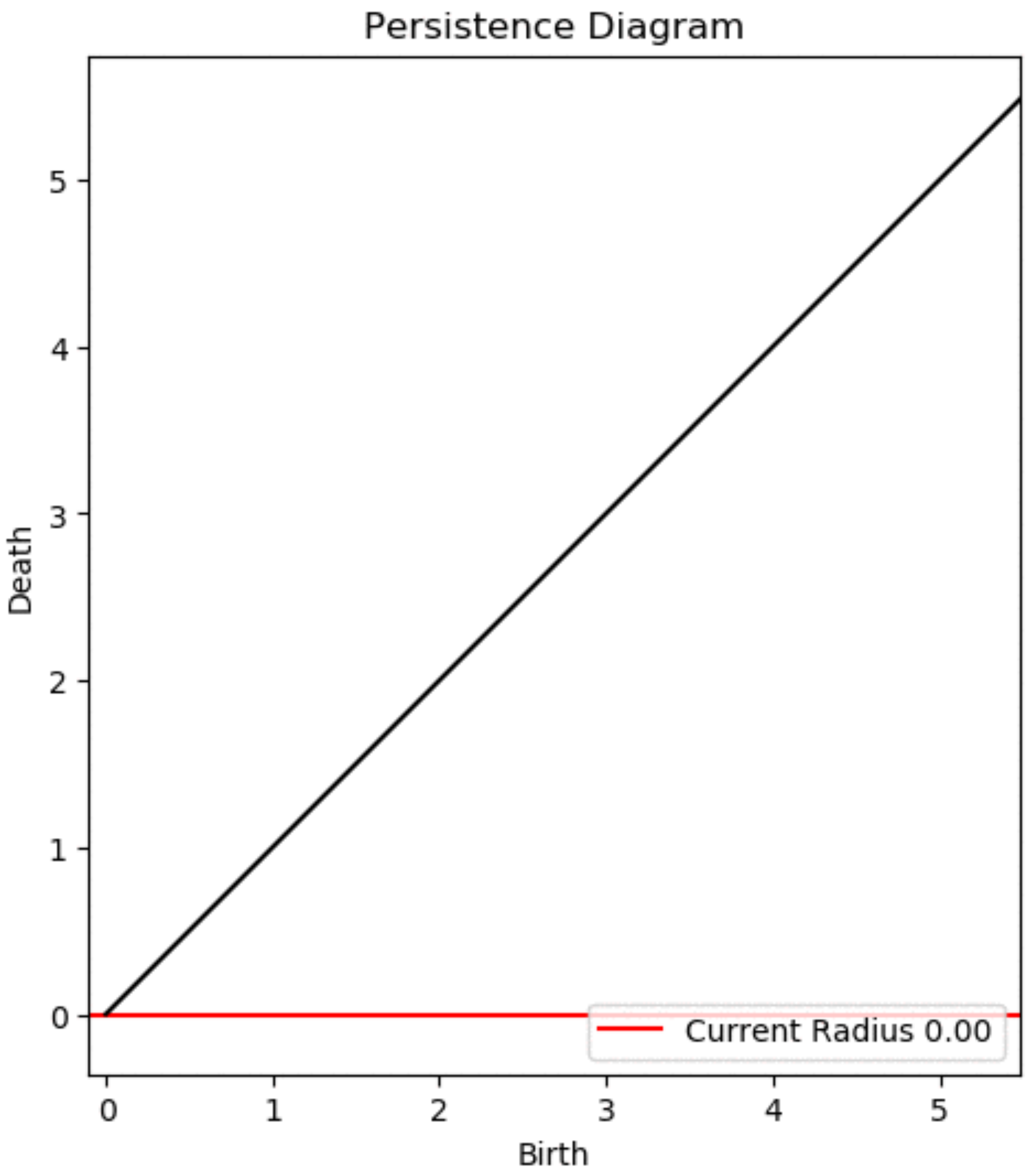
$H_1 = 0$

Persistent homology H_0

The first interesting threshold value is 0. At 0, a connected component for each point is born — each one of these is represented by a ball with none of the balls intersecting.

Persistent homology H_0





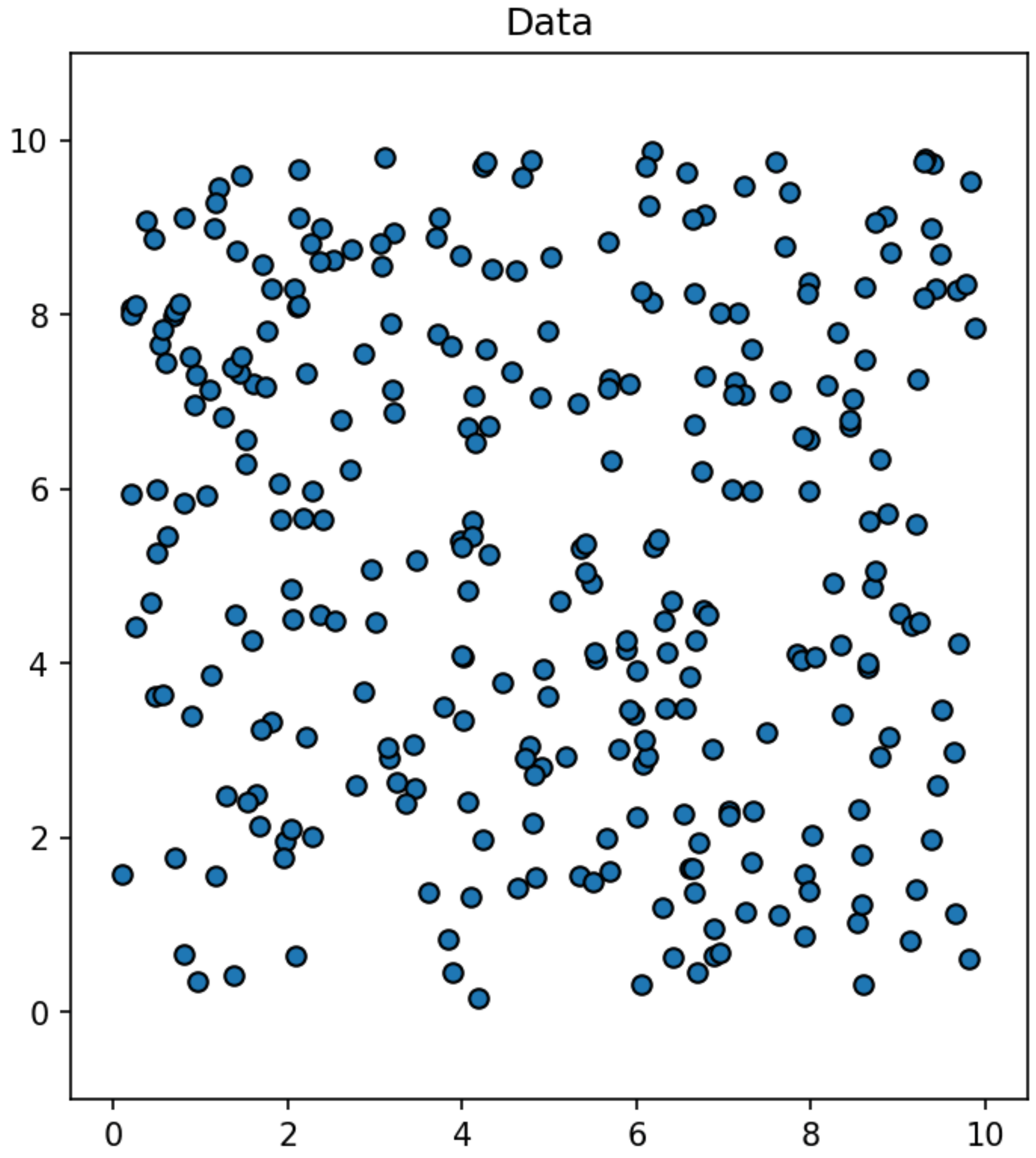
Persistent homology H_0

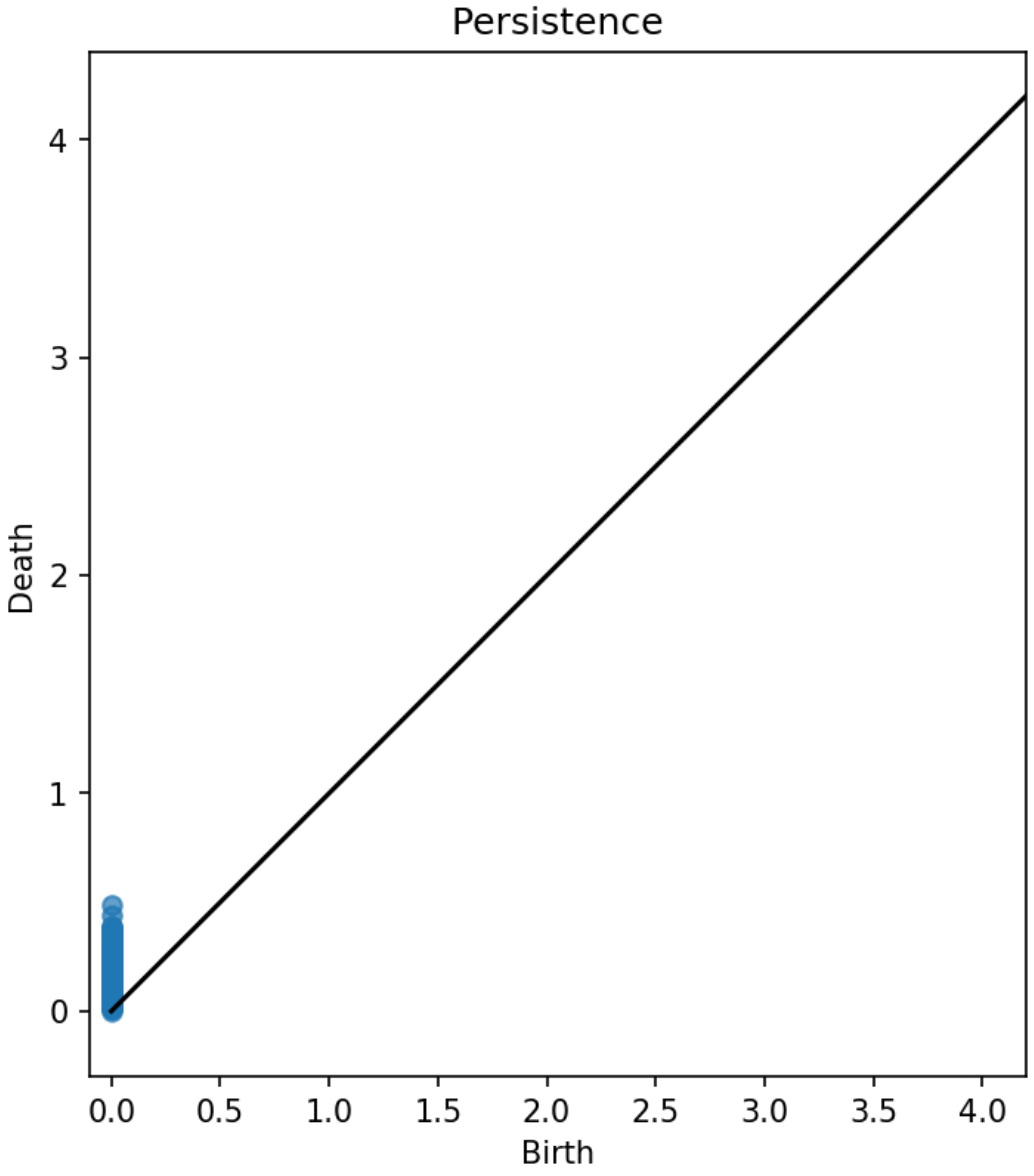
To the eye, it should be clear these data have some **semblance of two noisy clusters**. This leads to predictable effects on the persistence diagram. As the disks grow from 0 to 0.66, we see multiple (birth, death) pairs quickly appearing on the persistence diagram on the right. This should not be surprising--points close to each other quickly touch as each disk's radius increases.

Once we reach 0.66, however, we see on the left that the disks in each cluster are connected into **two disjoint clusters** (light blue and orange). Thus, these components have room to grow without touching a disjoint component, leading to no additional deaths for a while and thus a pause in new points appearing on the persistence diagram.

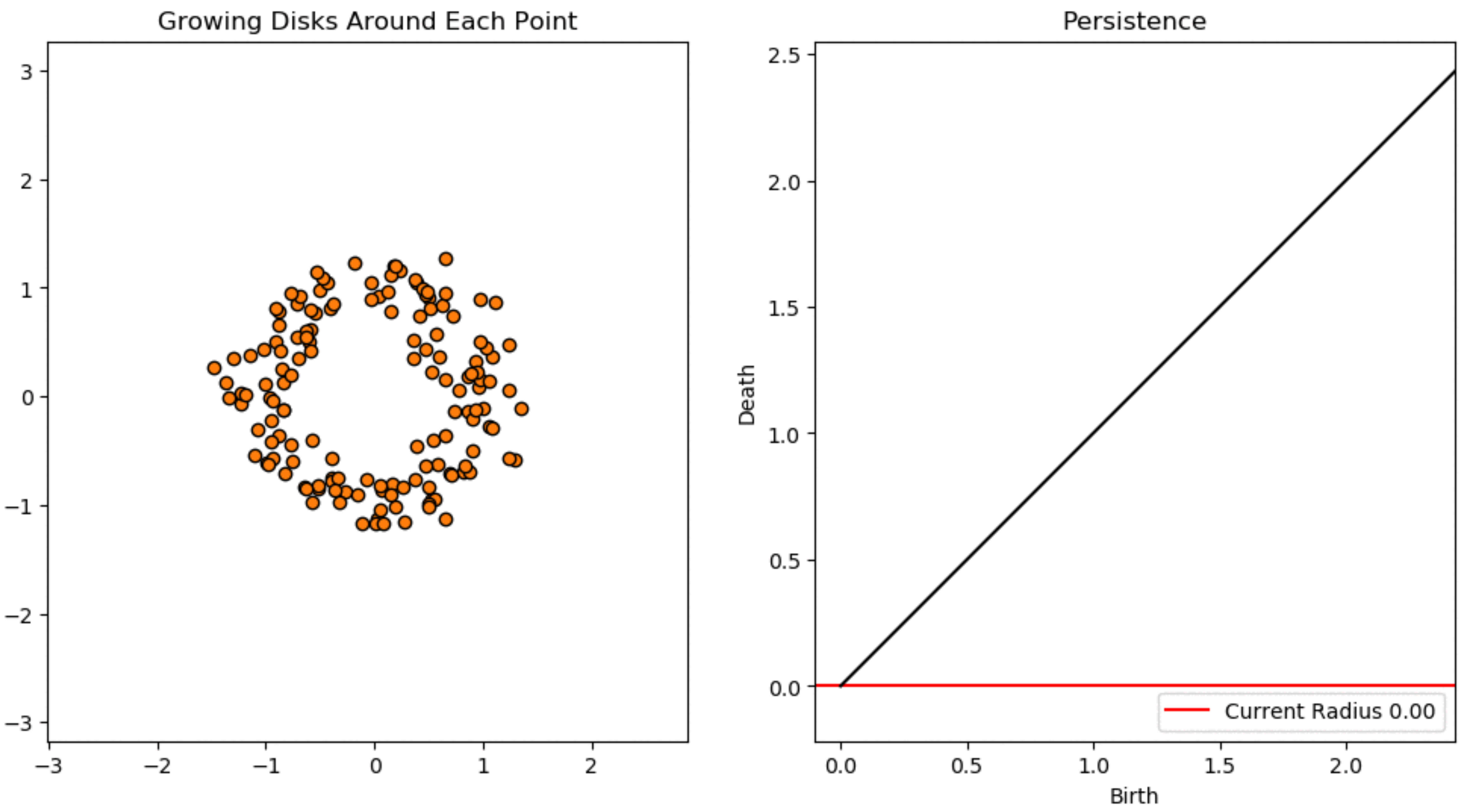
Similarly, **if we pushed the noisy clusters closer together, that gap would shrink.**

Persistent homology H_0

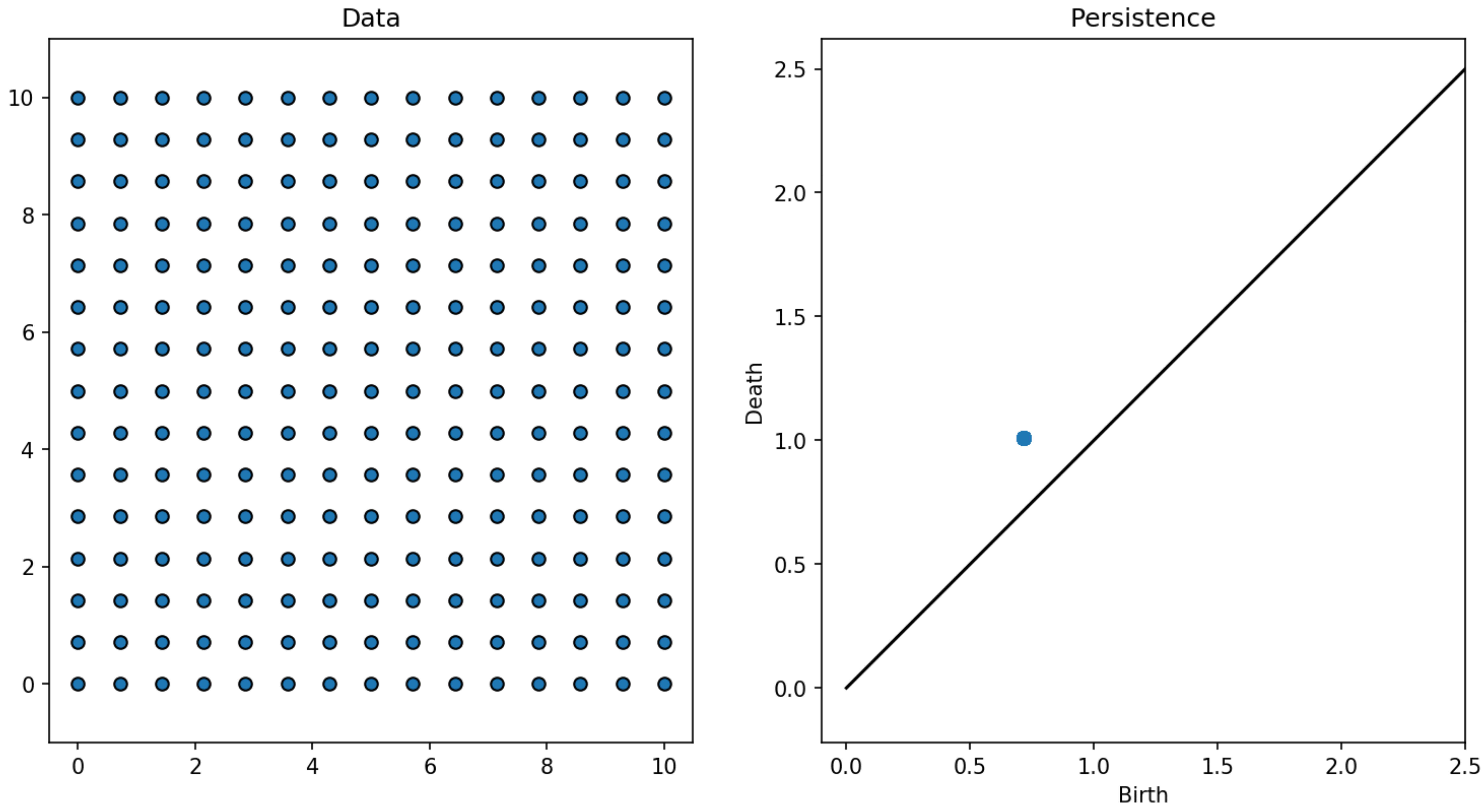




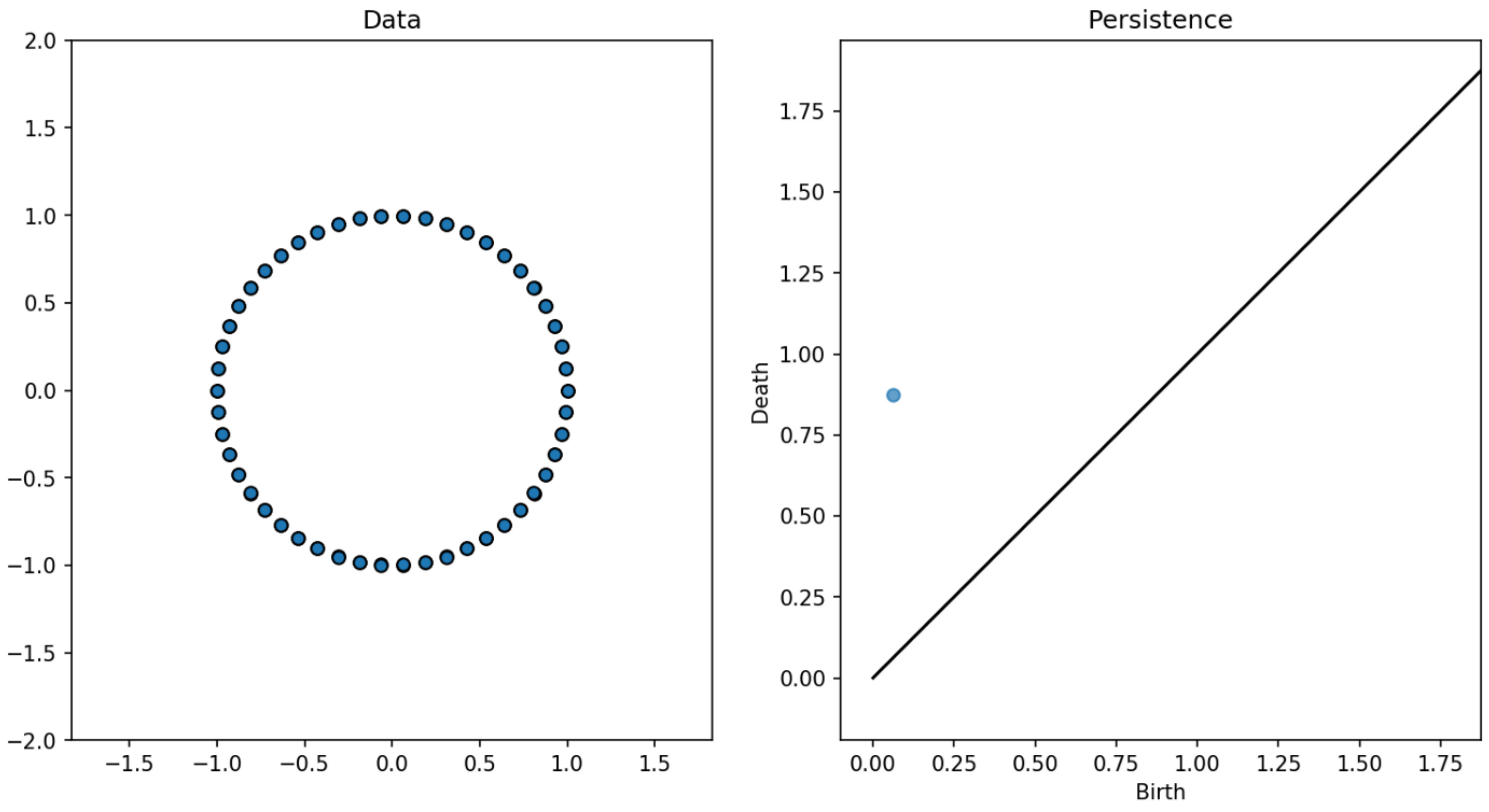
Persistent homology H_1

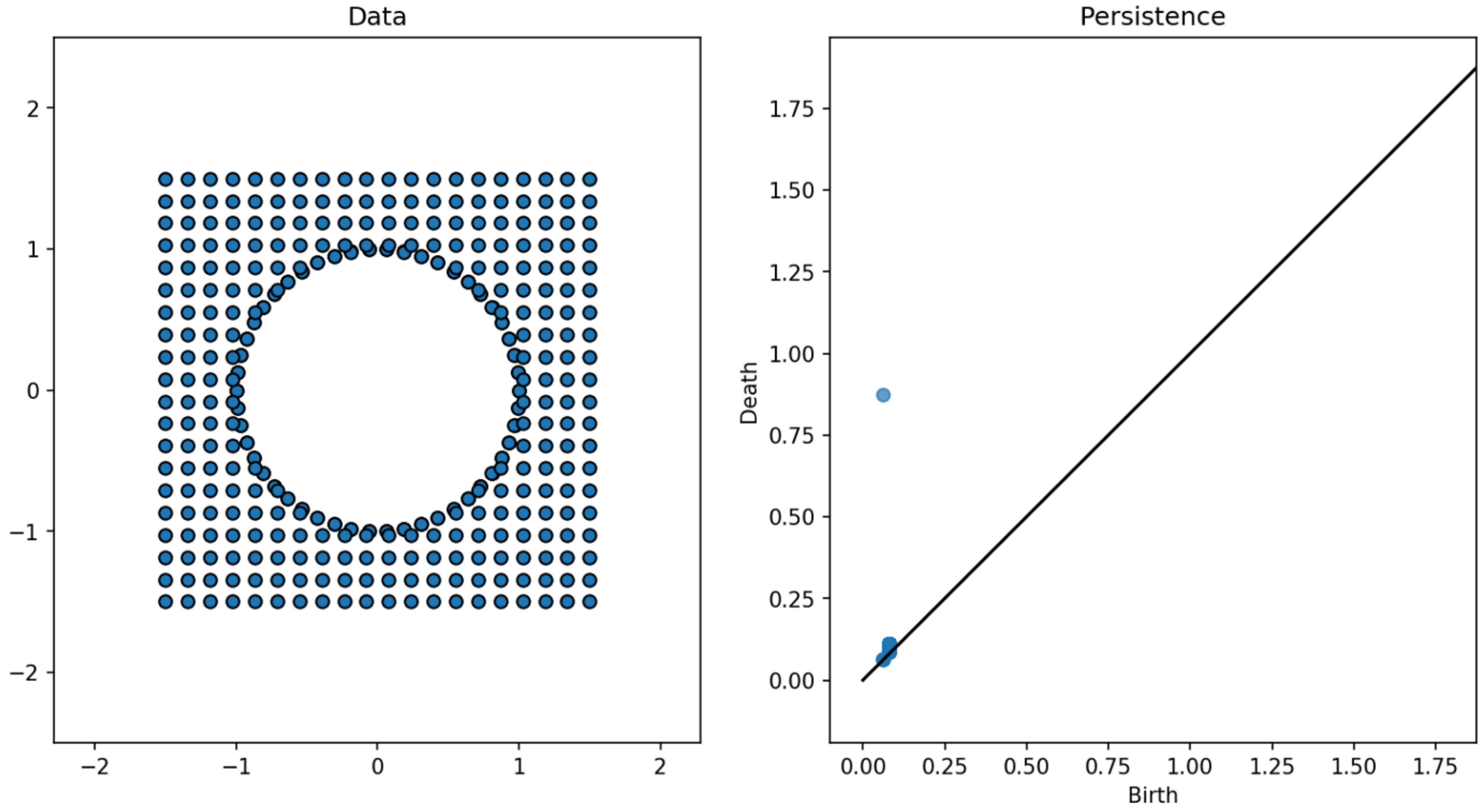


Persistent homology H_1

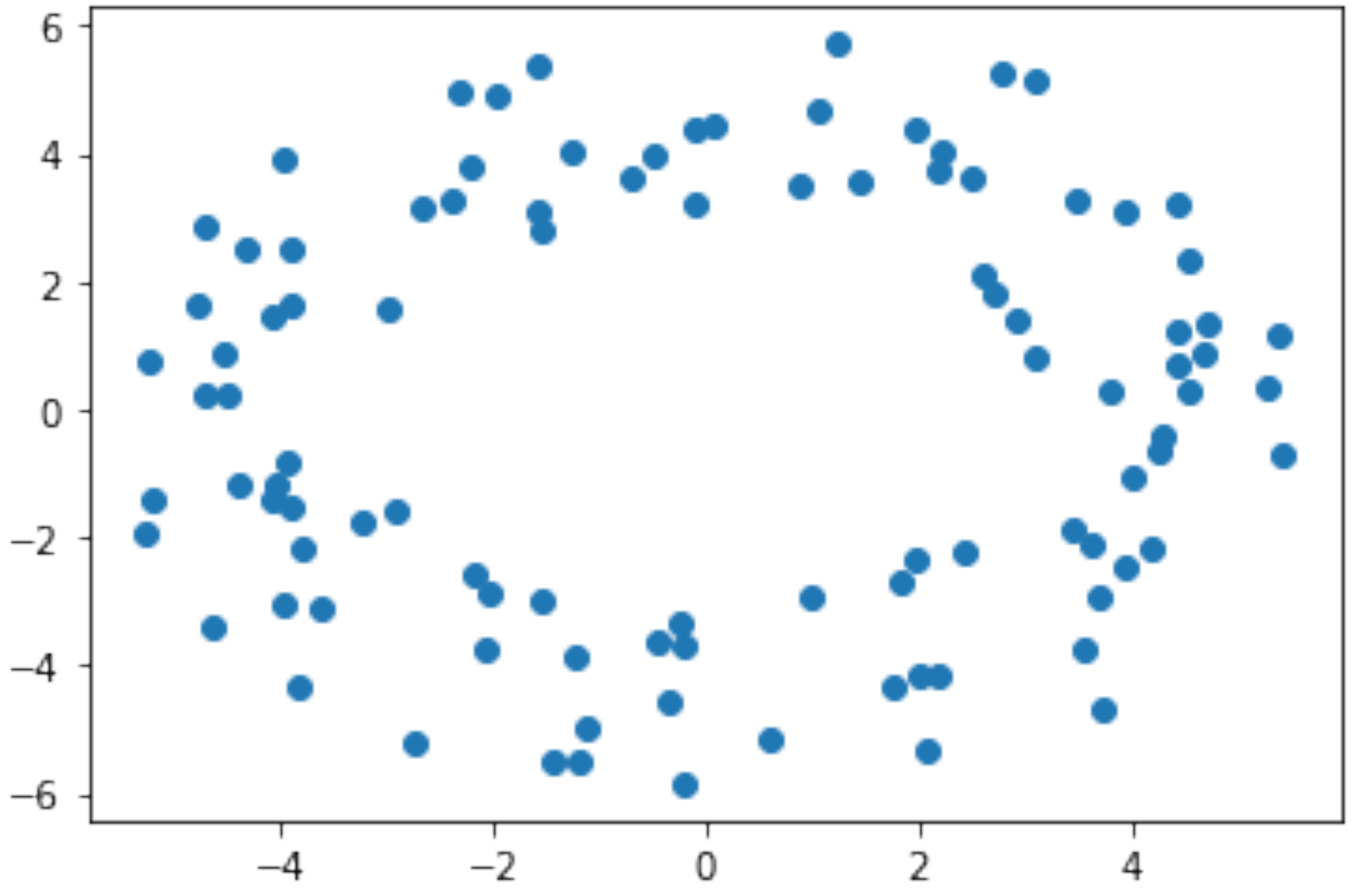


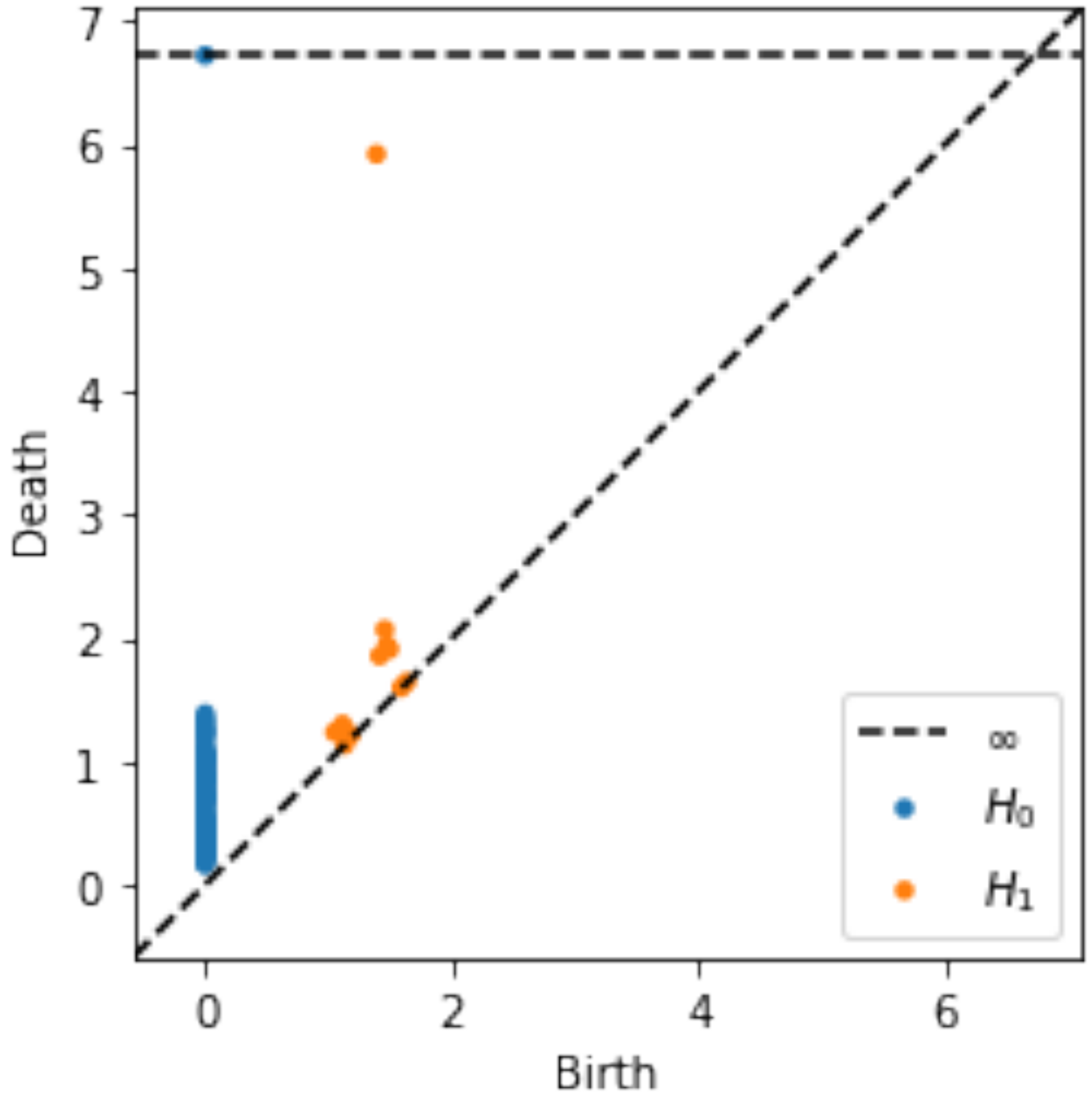
Persistent homology H_1



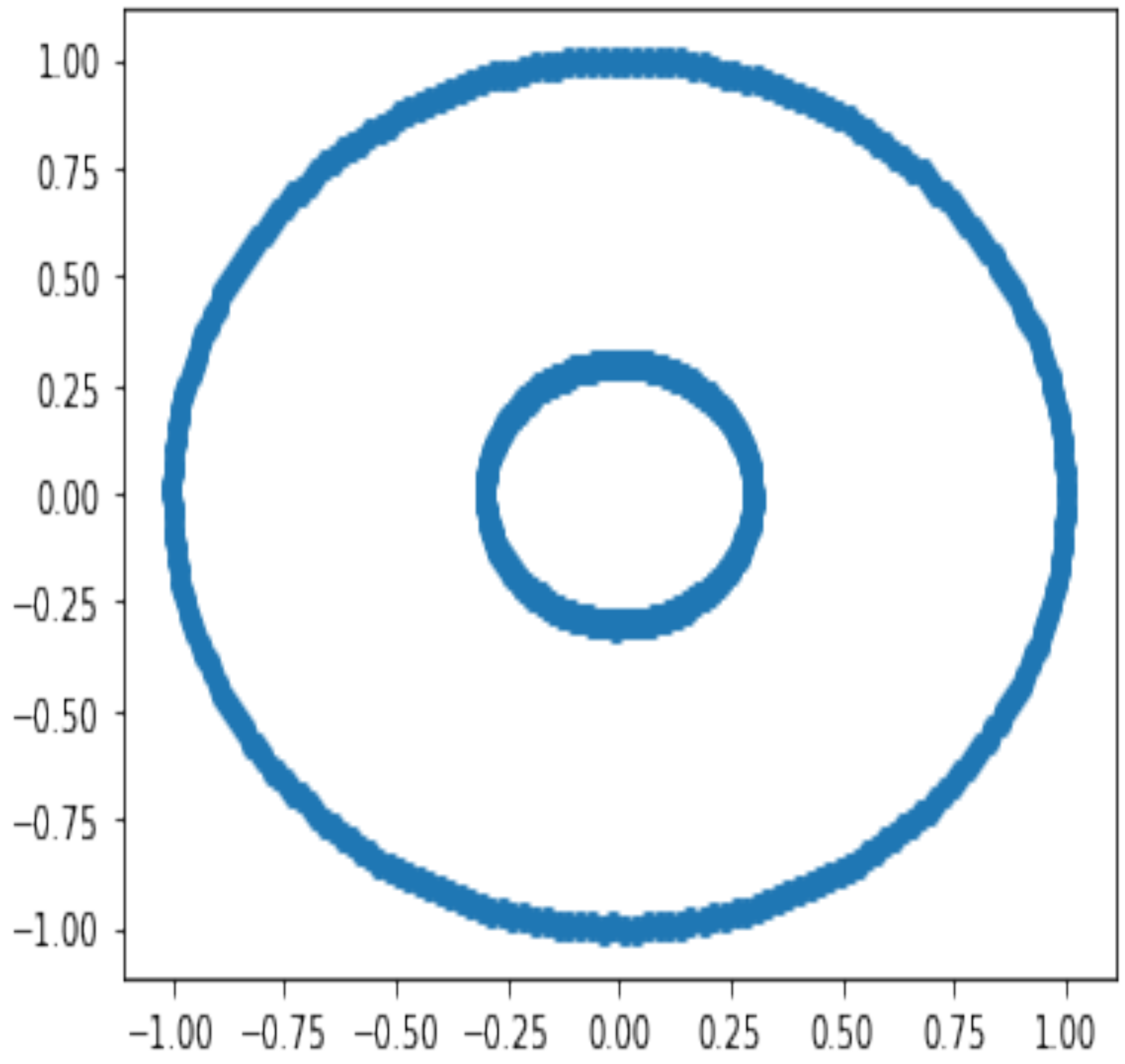


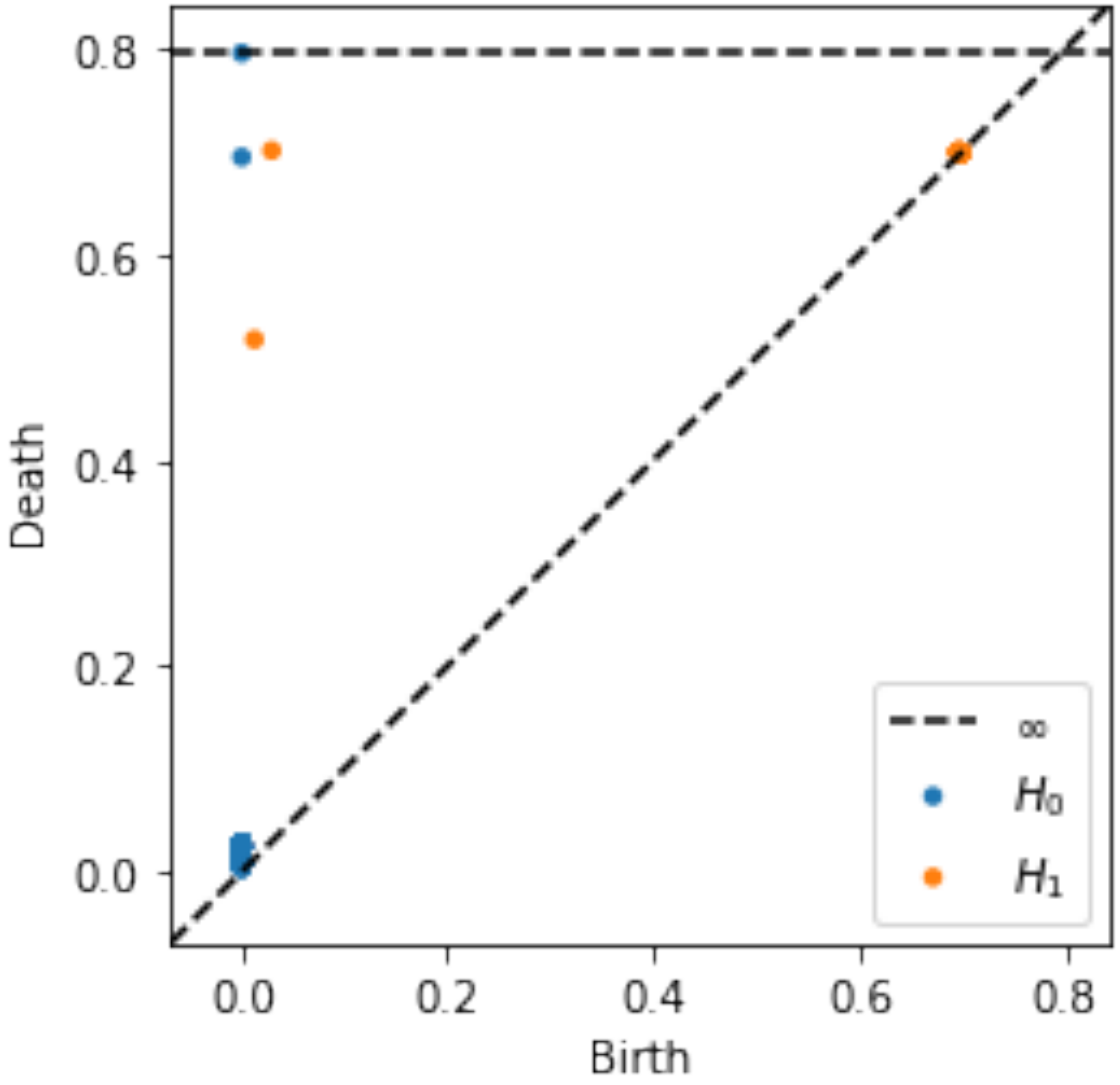
Practical example 1



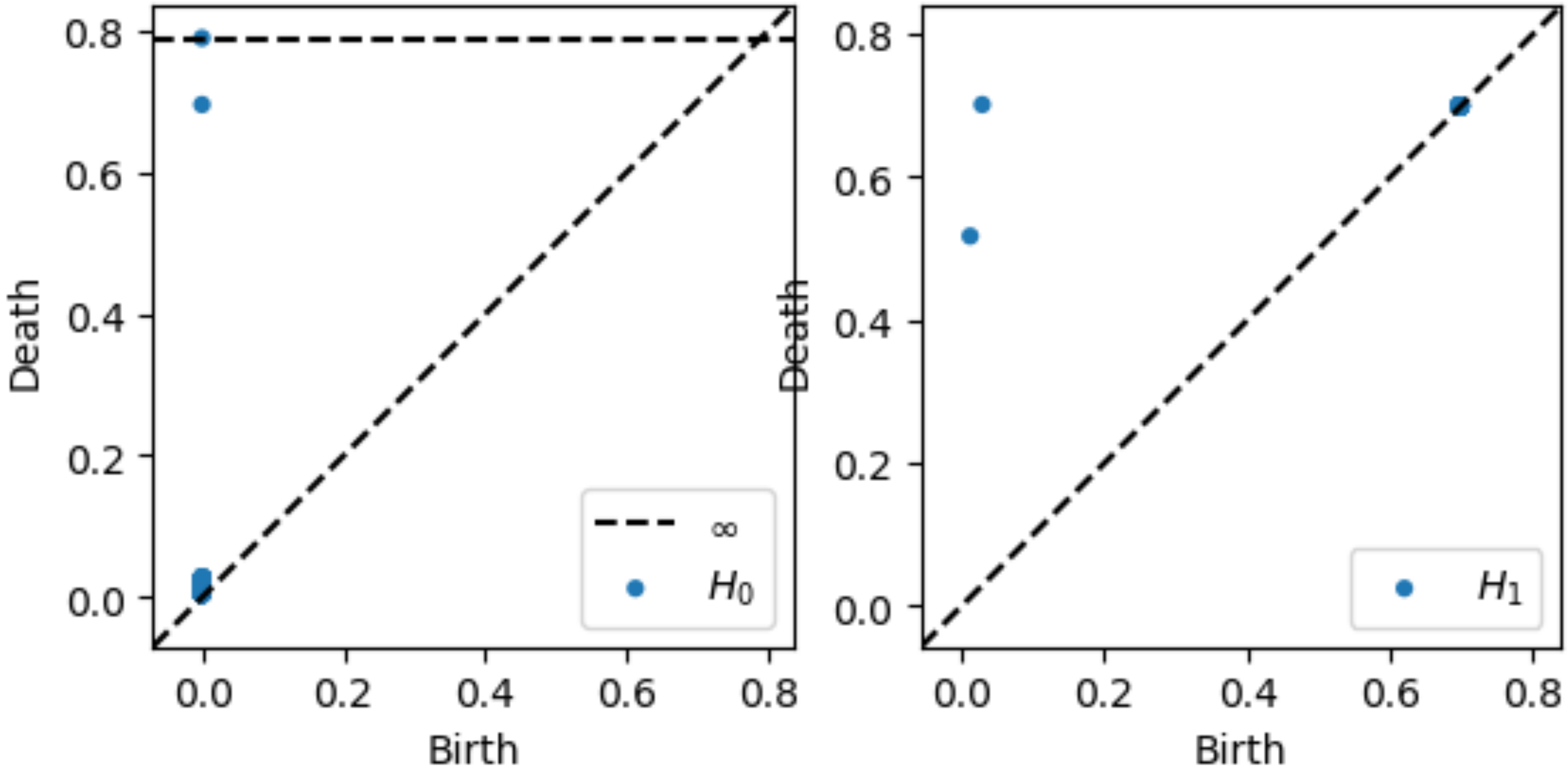


Practical example 1

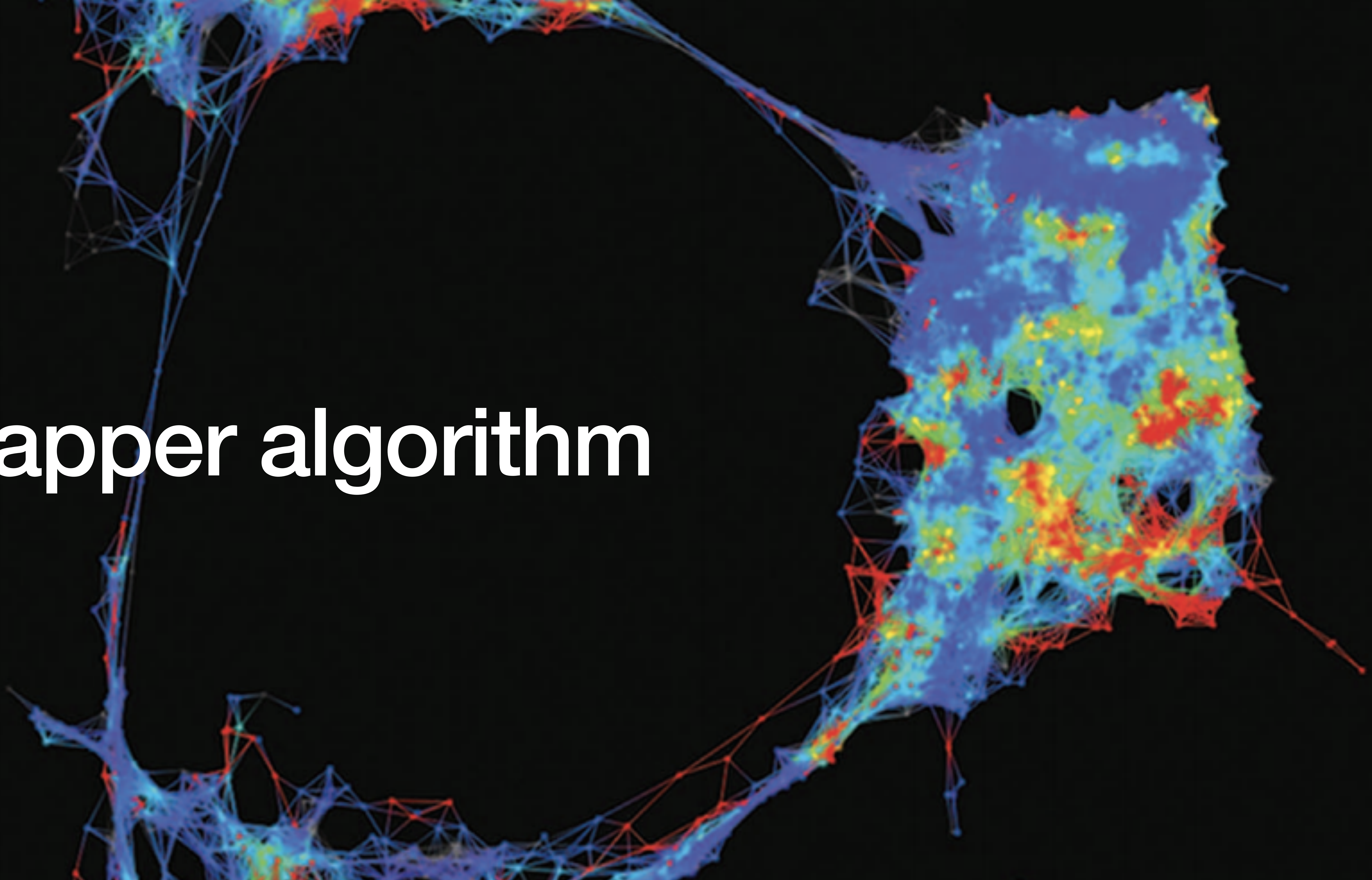




Practical example 1



Mapper algorithm



What is Mapper?

- As one of the main tools from the field of Topological Data Analysis, **Mapper** has been shown to be particularly useful for exploring high dimensional point cloud data.
- **Mapper** is way to construct a graph (or simplicial complex) from data in a way that reveals the some of the topological features of the space.
- Is an unsupervised method of generating a visual representation of the data that can often reveal new insights of the data that other methods cannot. Most importantly, once constructed **Mapper** can be used by nonexperts to explore the structure of a data set or function on the data.

What is Mapper?

The formal definition of **mapper** is a **simplicial complex** constructed by taking the **nerve** of the **refined pullback** of an overlapping cover of a **lens function**. This definition boils down to simply splitting up the data points into buckets, clustering the points within each bucket, and then stringing the clusters together into a graph.

To construct the mapper, we need to define two pieces

1. a lens
2. a cover of the lens

The lens is a function through which we observe the data. It is a summarization of your data in a much lower dimensional space.

What is Mapper?

The formal definition of **mapper** is a **simplicial complex** constructed by taking the **nerve** of the **refined pullback** of an overlapping cover of a **lens function**. This definition boils down to simply splitting up the data points into buckets, clustering the points within each bucket, and then stringing the clusters together into a graph.

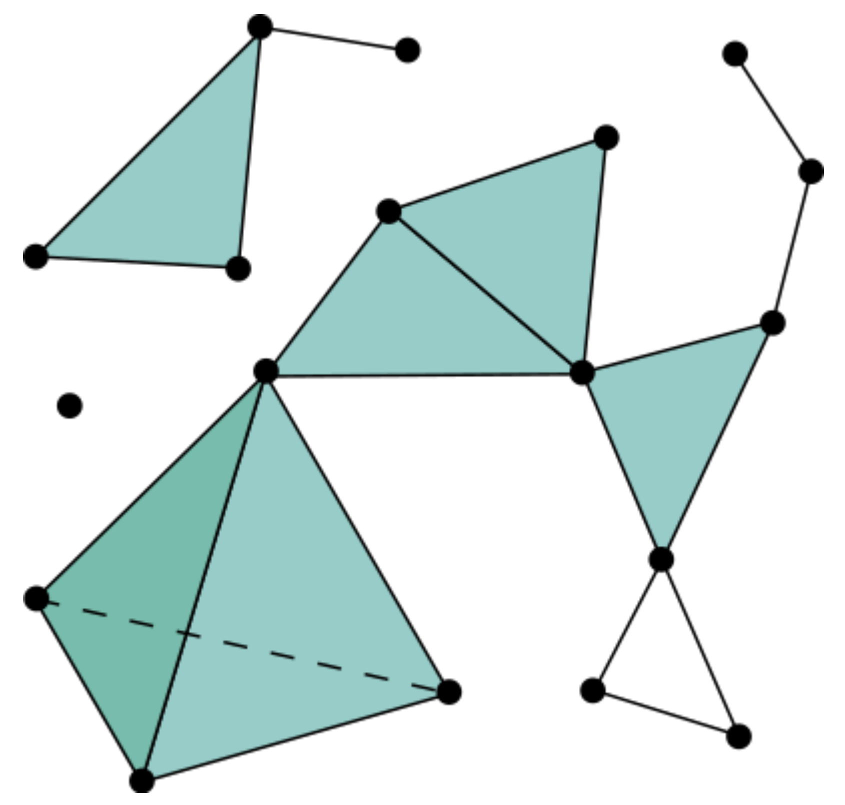
To construct the mapper, we need to define two pieces

1. a lens
2. a cover of the lens

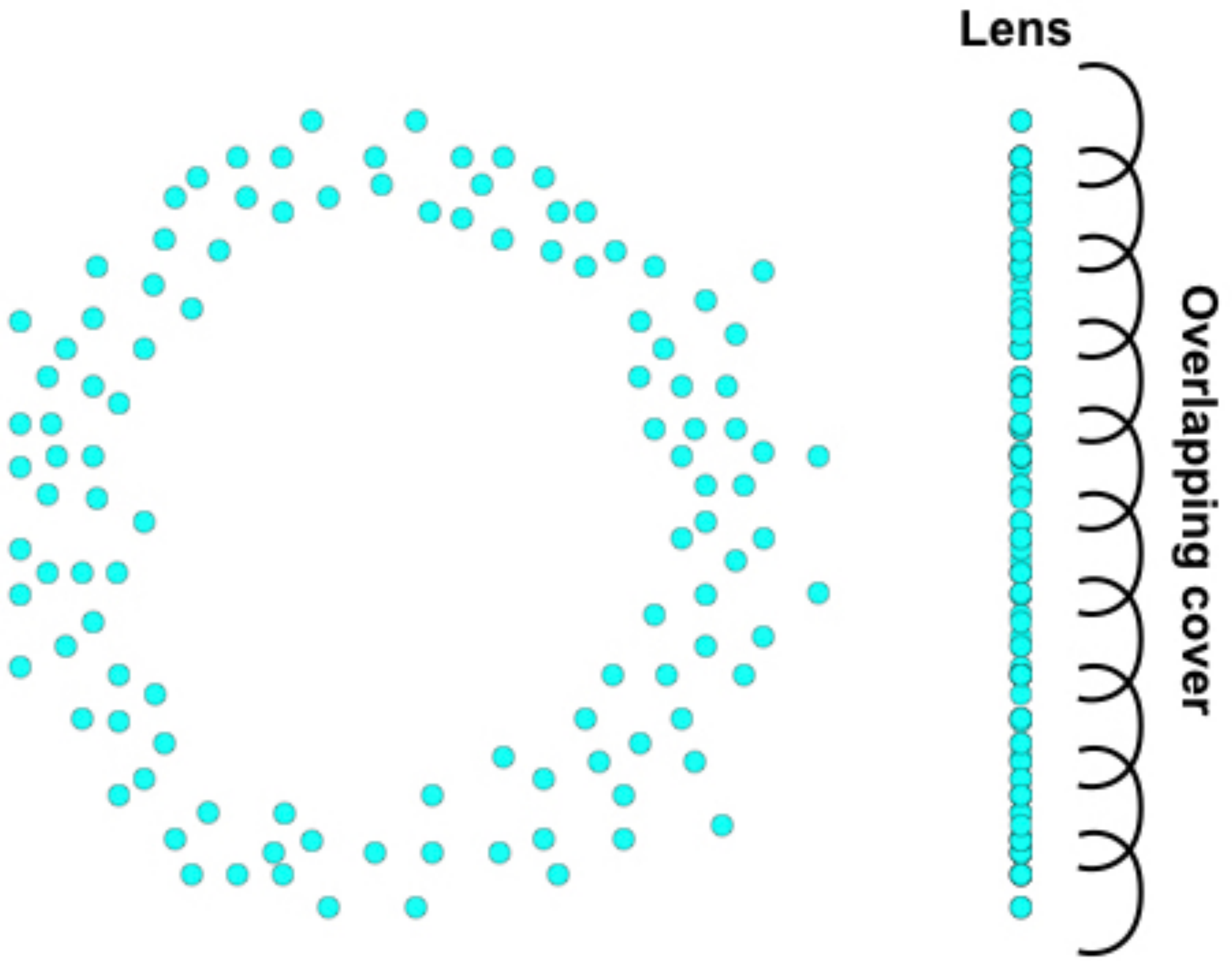
The lens is a function through which we observe the data. It is a summarization of your data in a much lower dimensional space.

Simplical complex

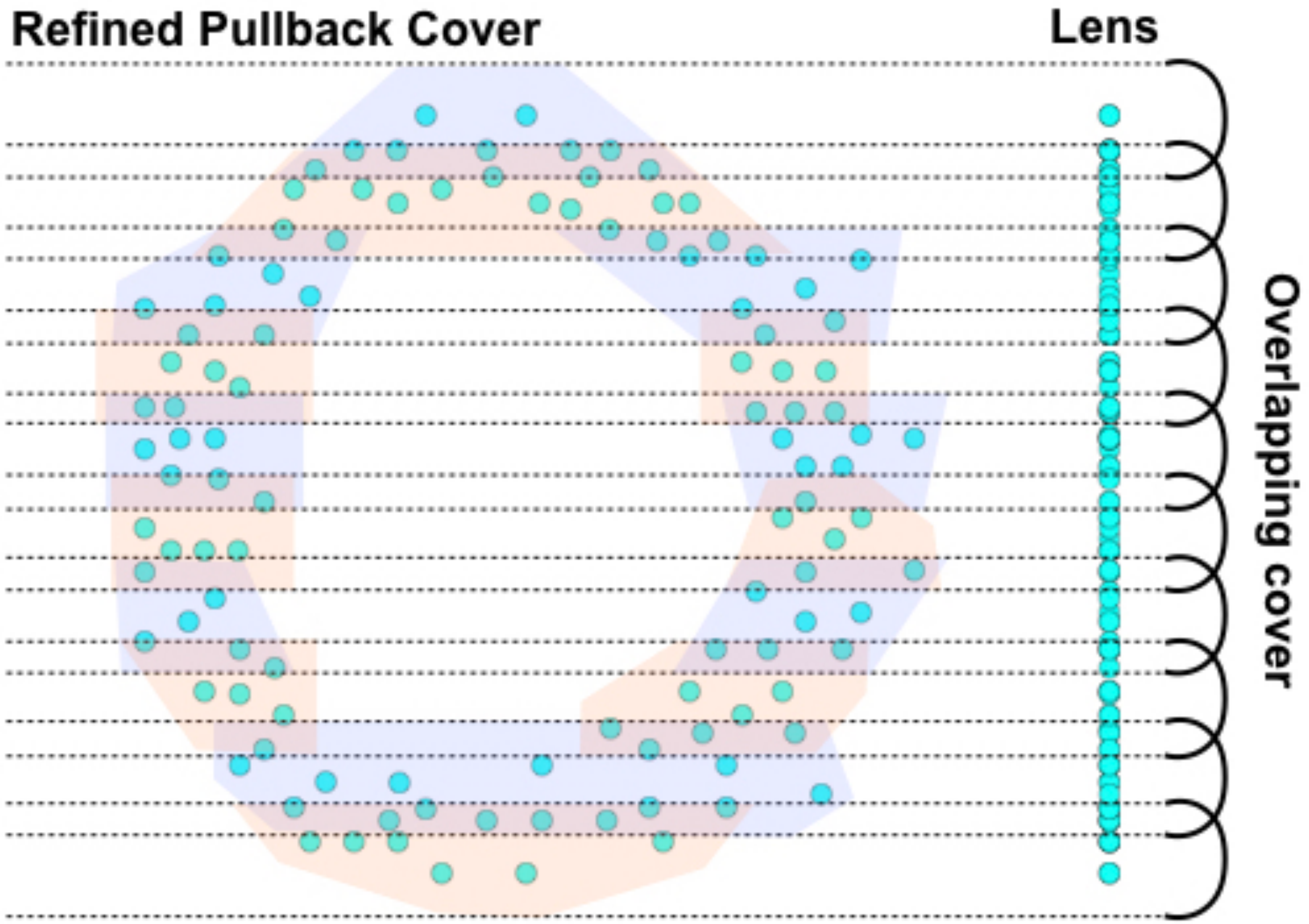
In mathematics, a simplicial complex is a set composed of points, line segments, triangles, and their n-dimensional counterparts



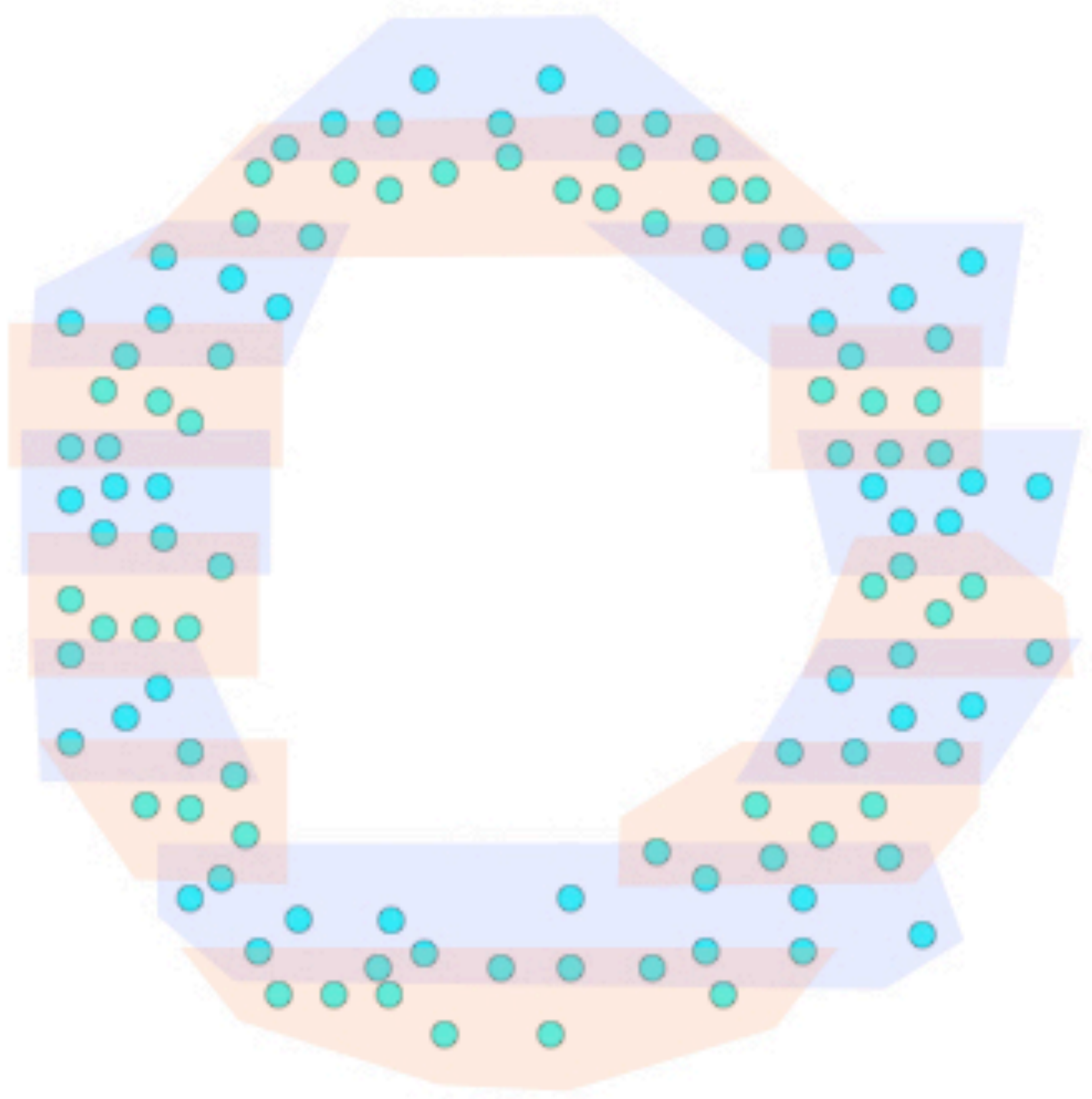
What is Mapper?

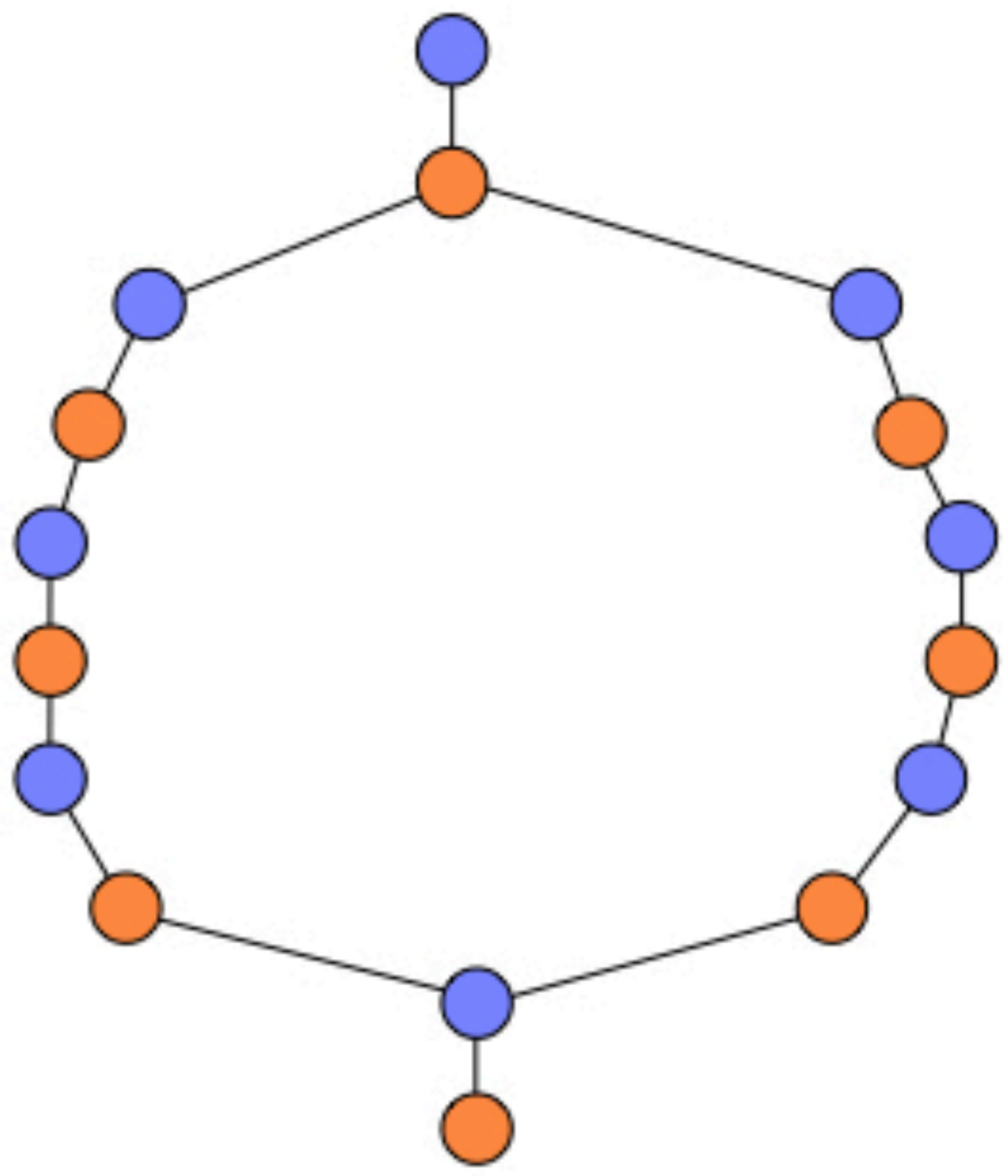


What is Mapper?

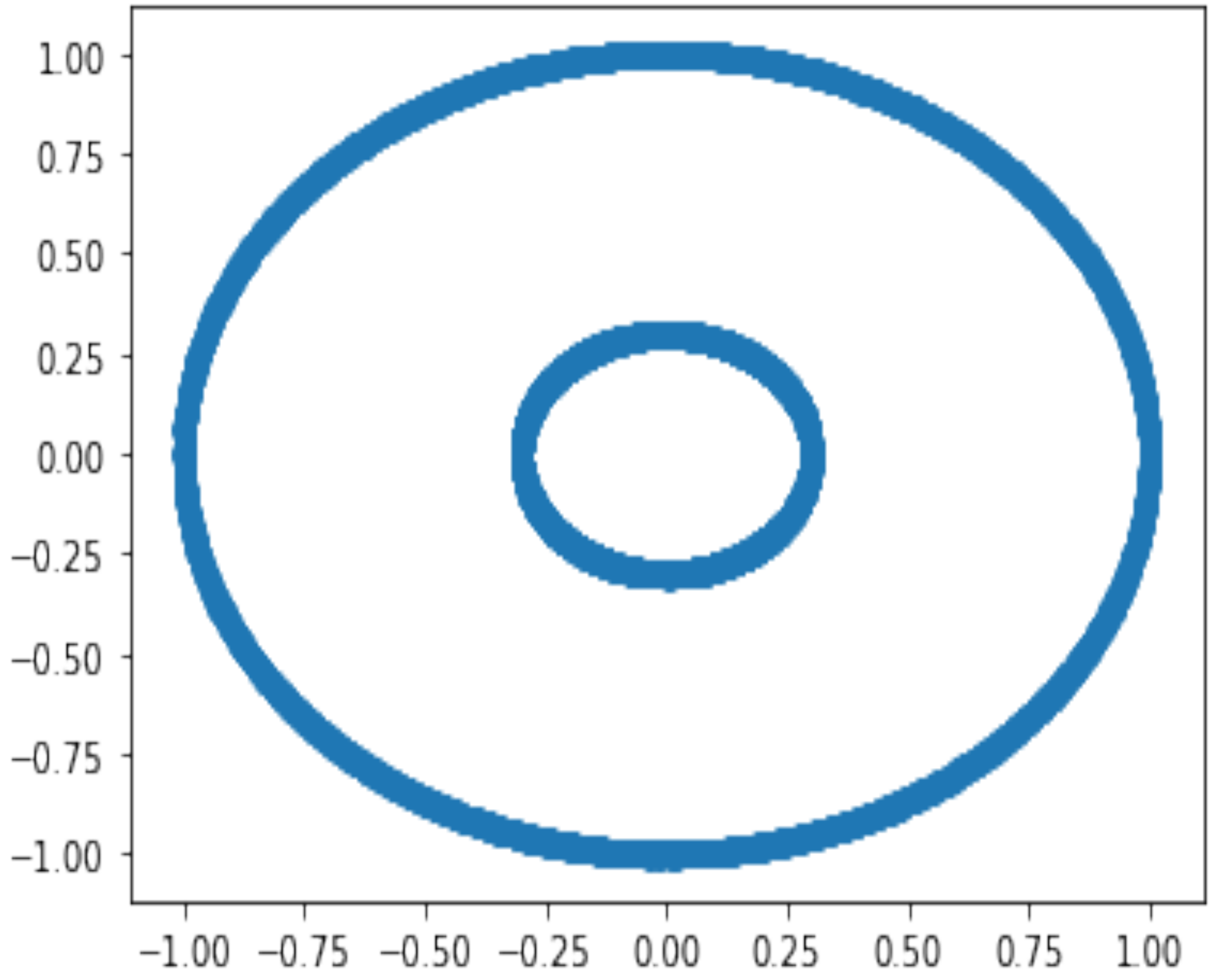


What is Mapper?

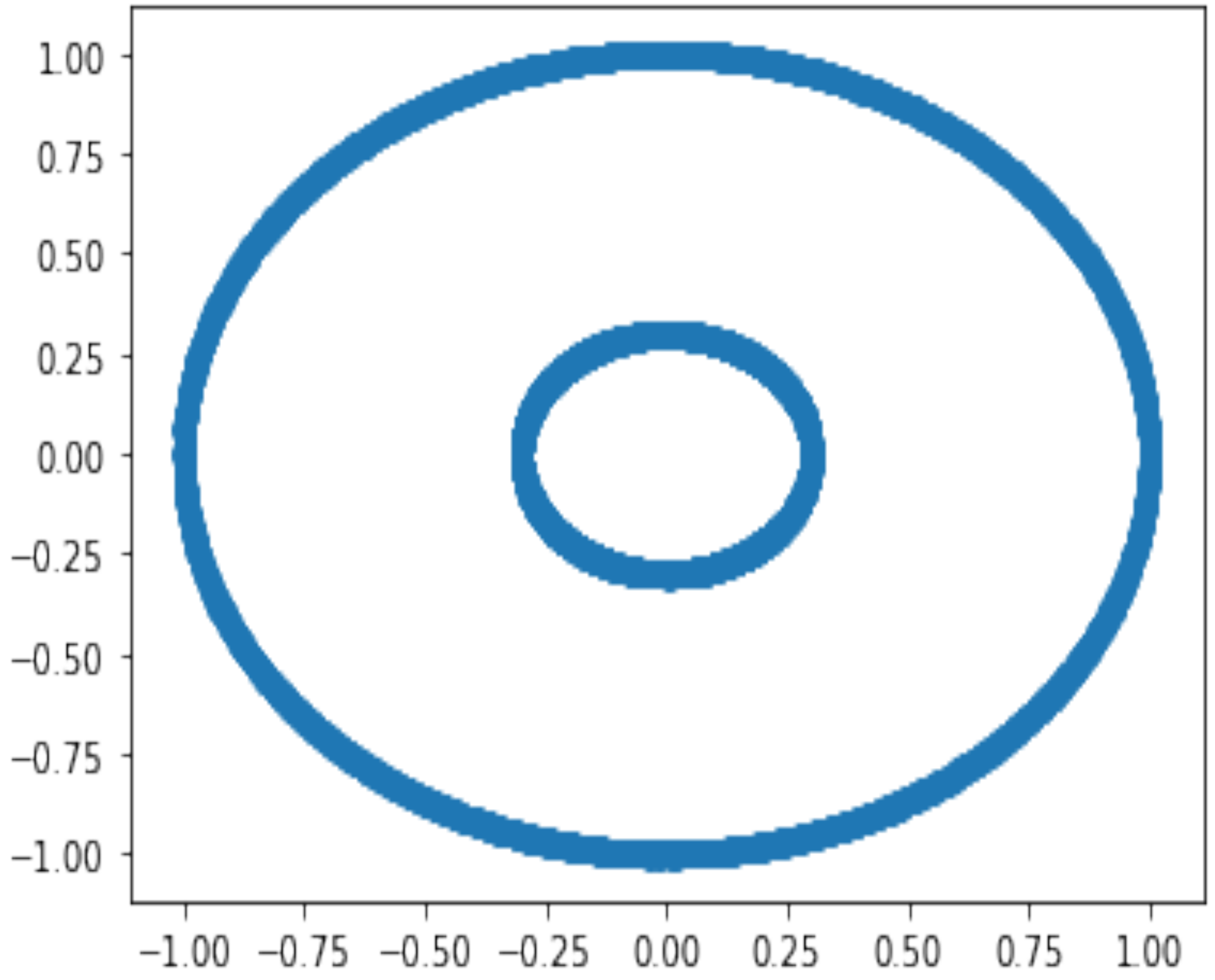




Example 1: Two circles

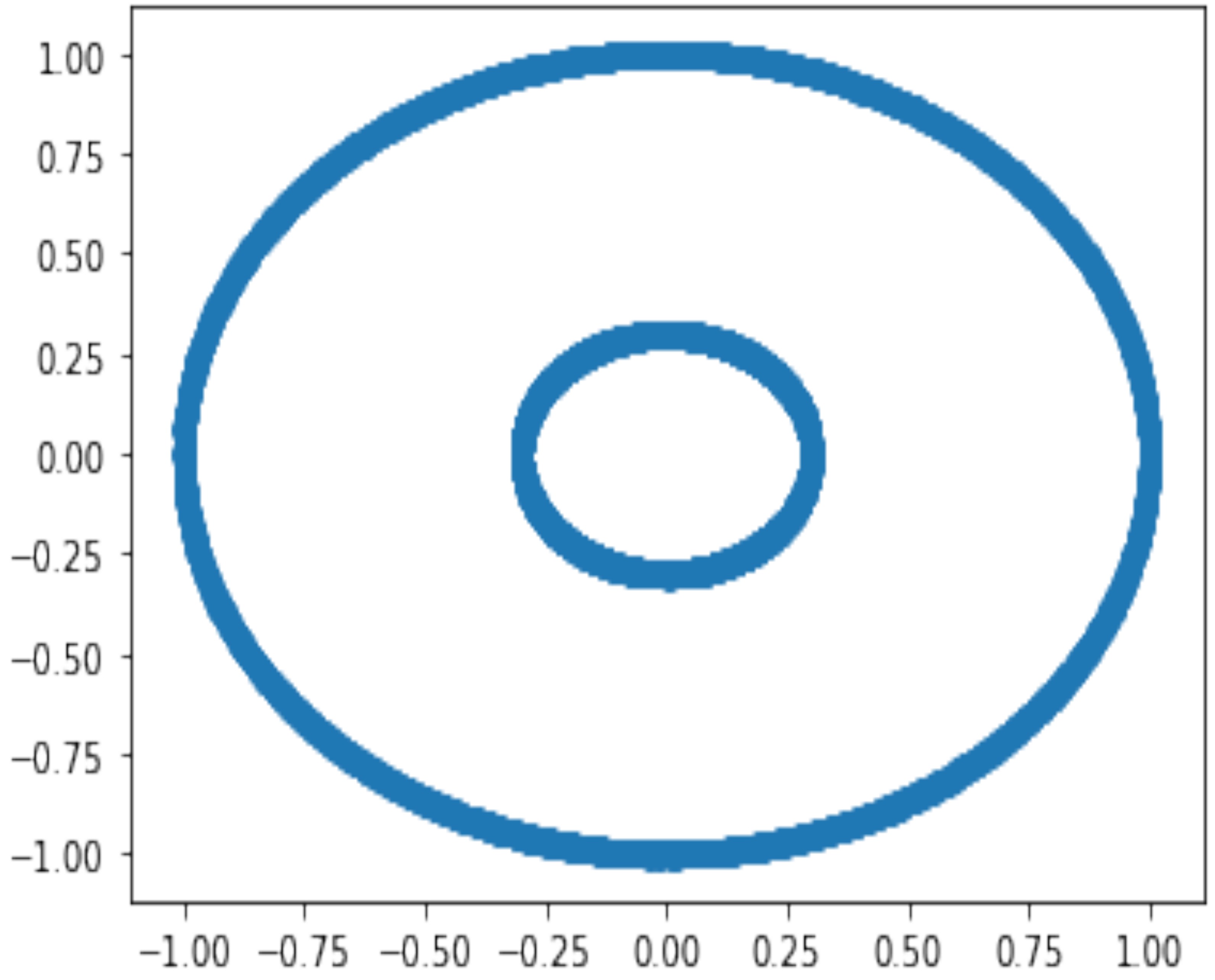


Example 1: Two circles

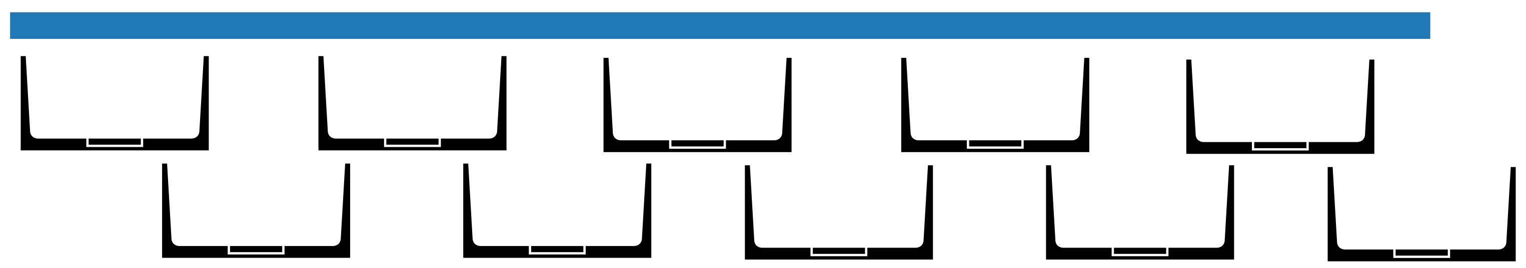


Lens: x projection

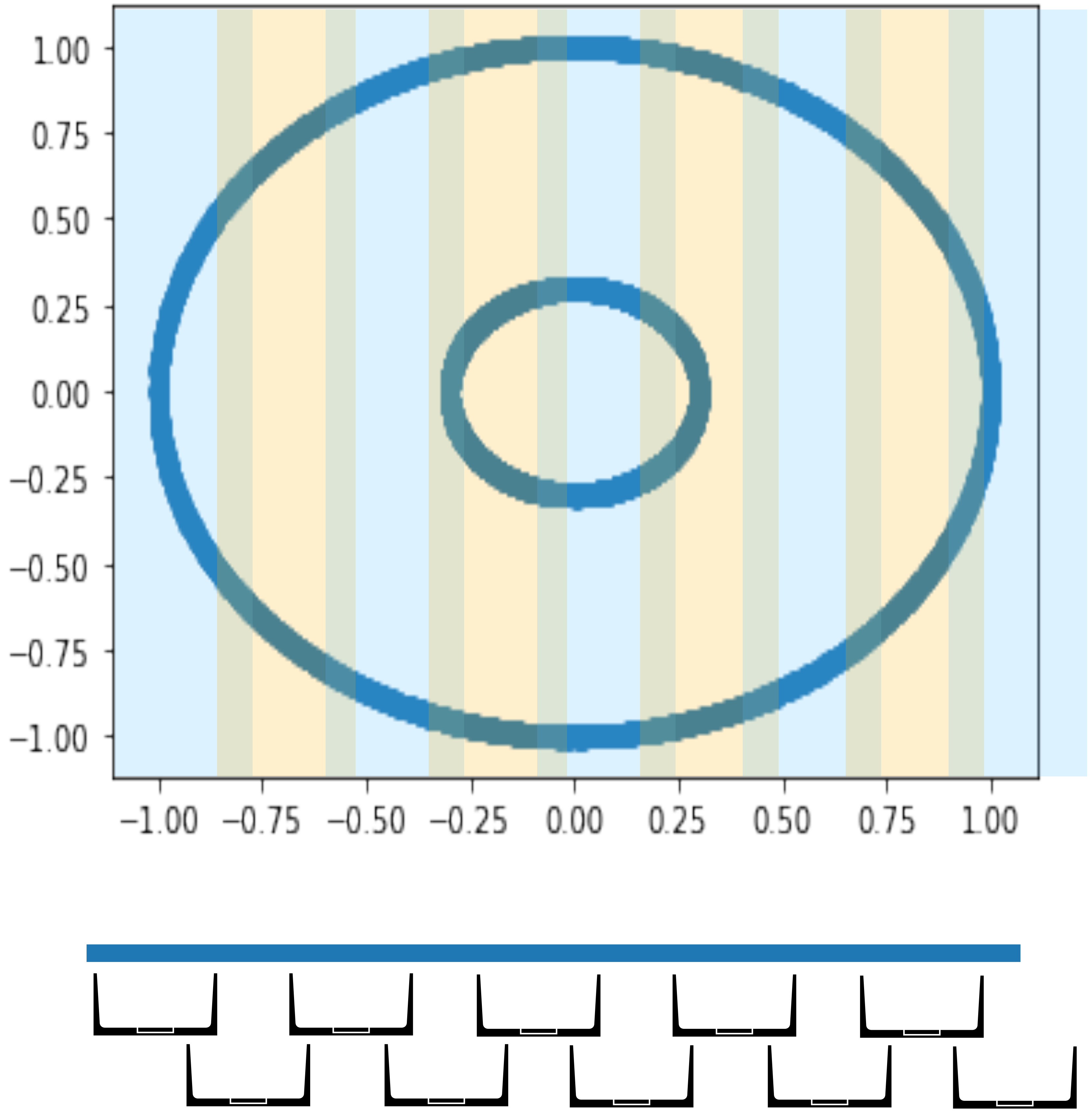
Example 1: Two circles



Lens: x projection



Example 1: Two circles



Lens: x projection

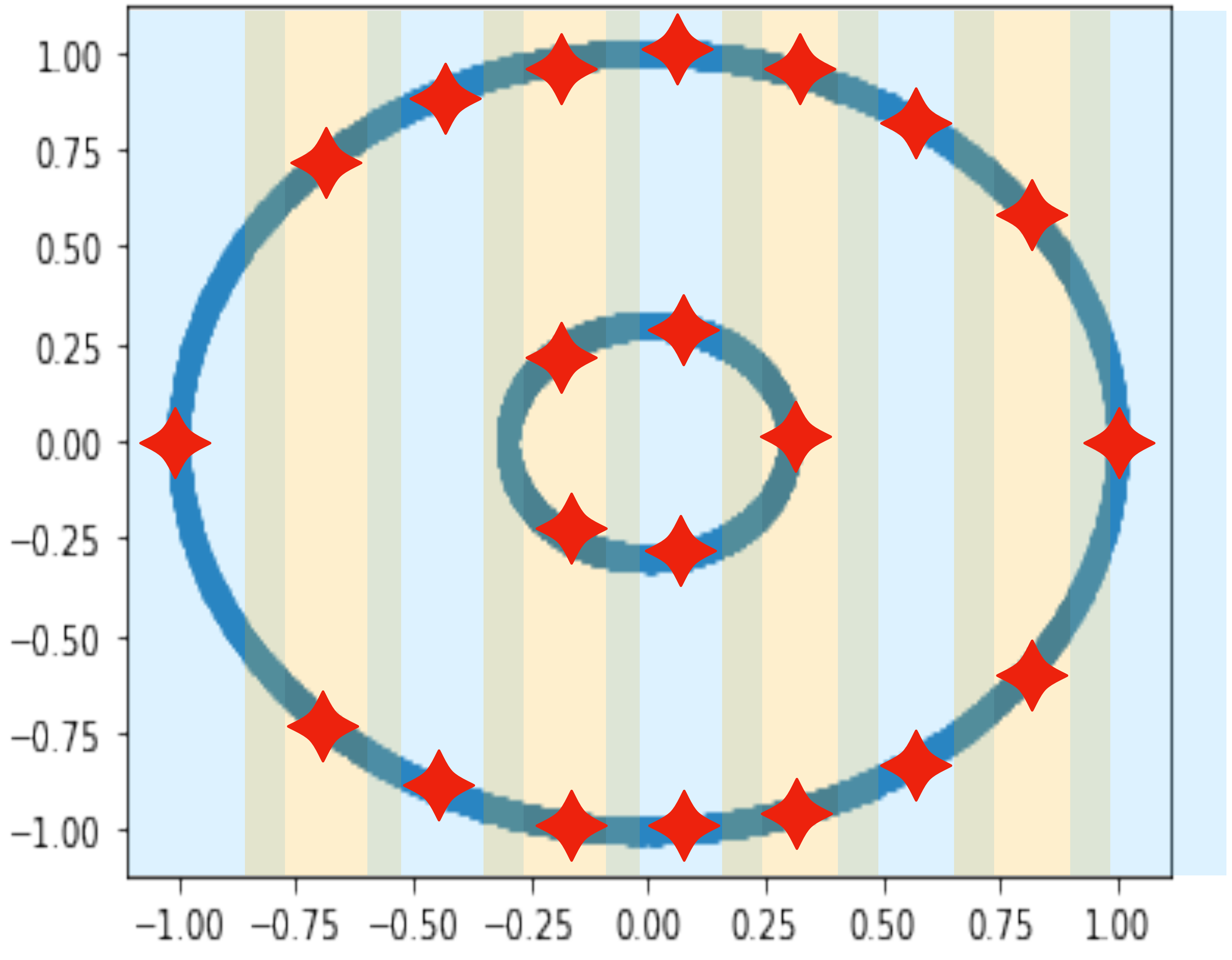
N = 10

Overlap = 0.35

Example 1: Two circles

Cluster algoritm:

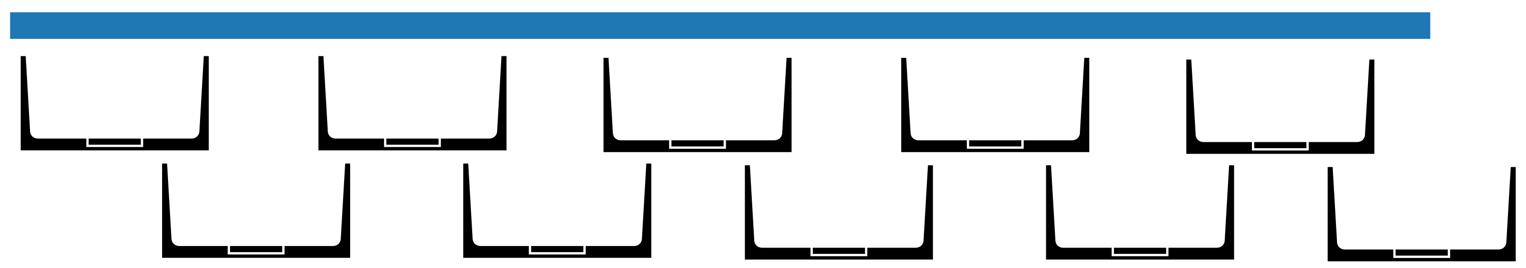
DBSCAN



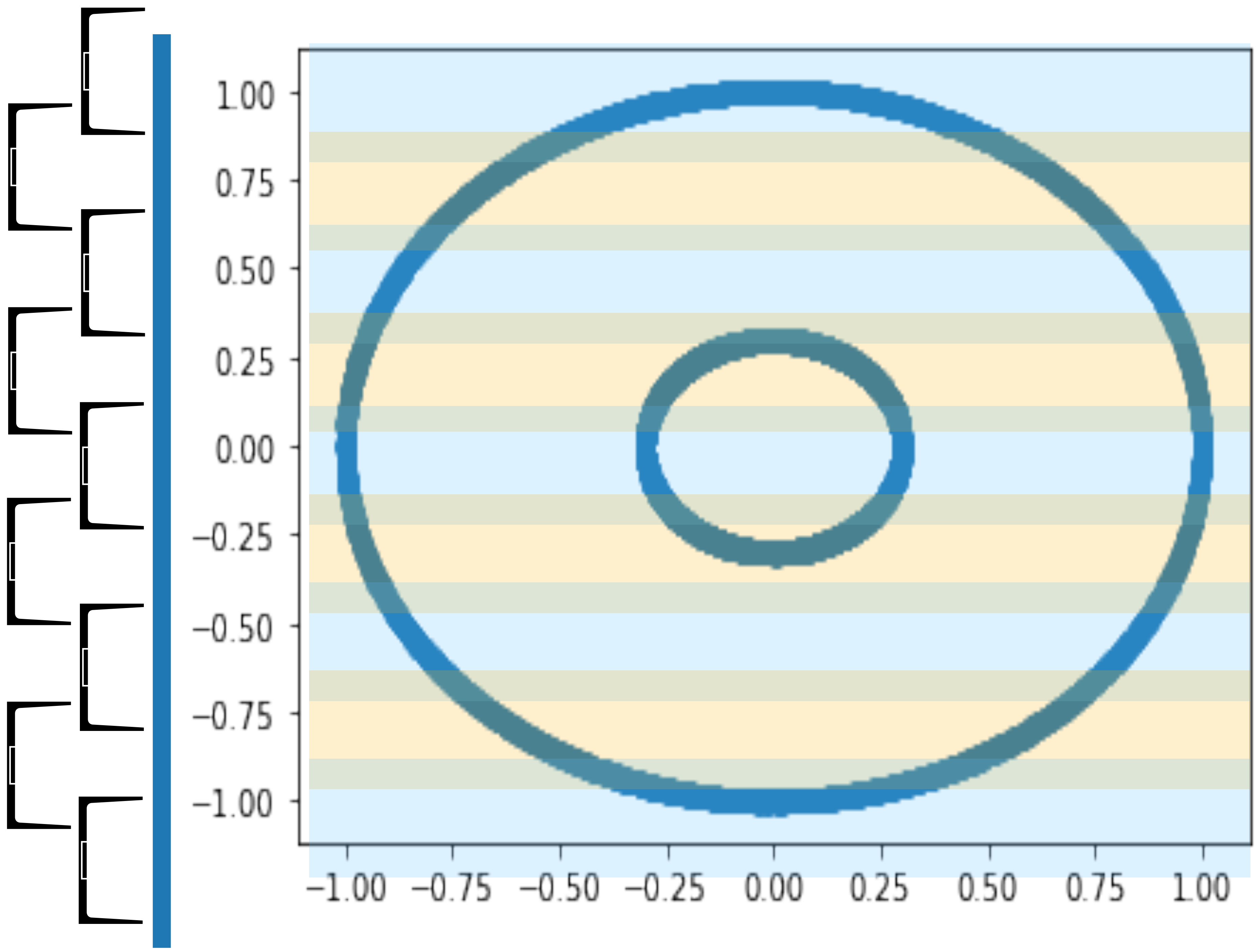
N = 10

Overlap = 0.35

Lens: x projection



Example 1: Two circles

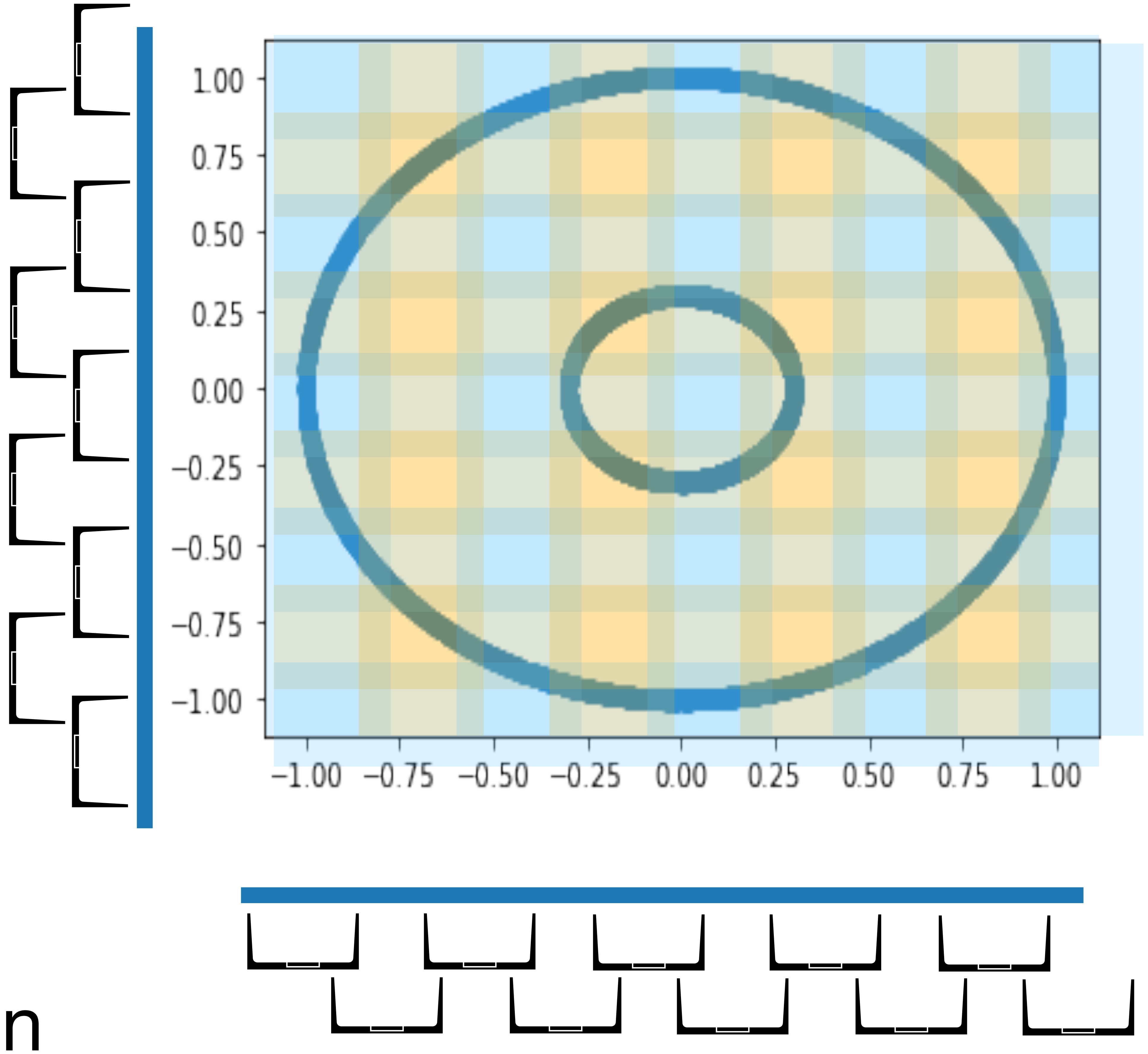


Lens: y projection

N = 10

Overlap = 0.35

Example 1: Two circles

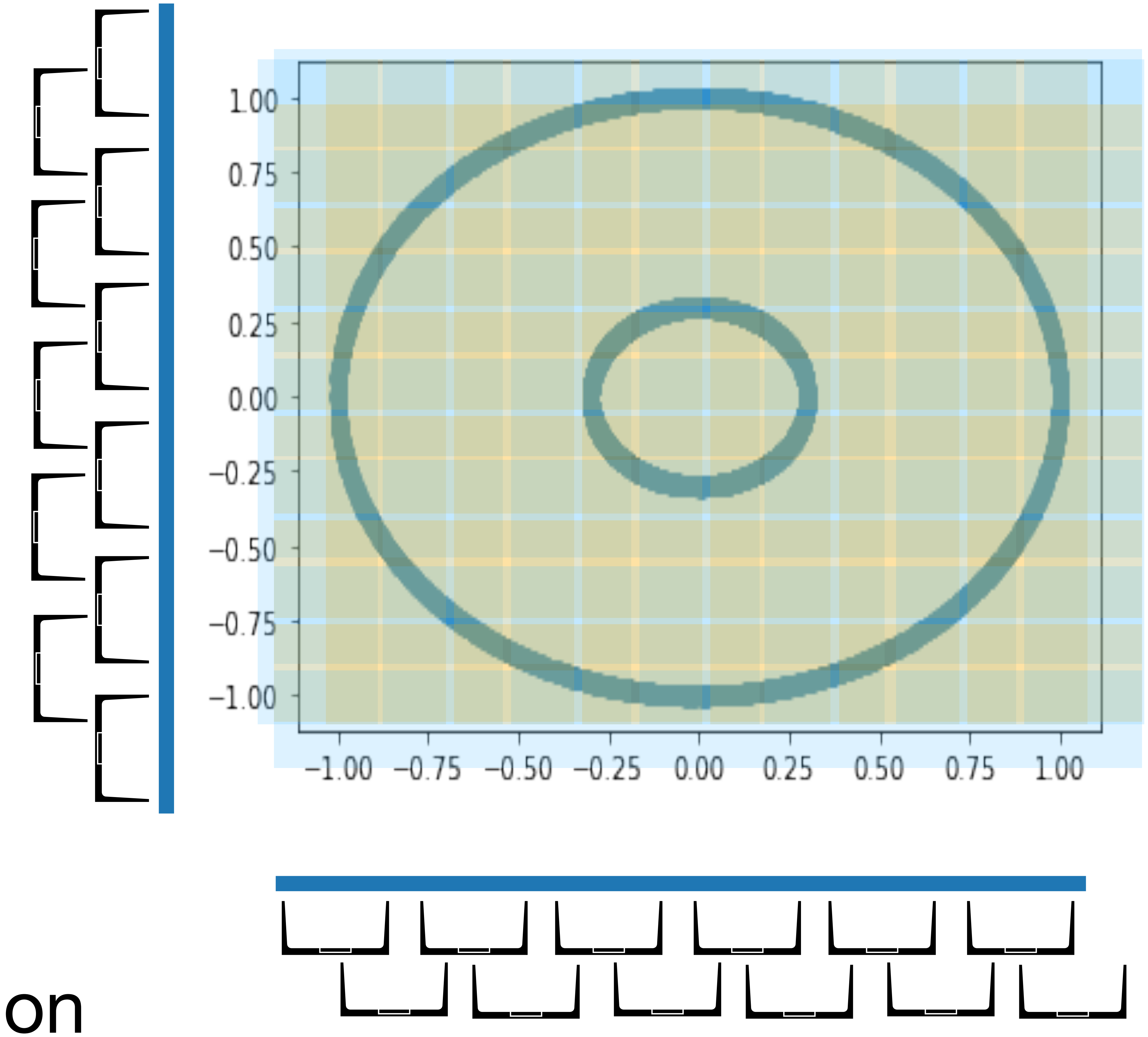


N = 10x10

Overlap = 0.35

Lens: x-y projection

Example 1: Two circles

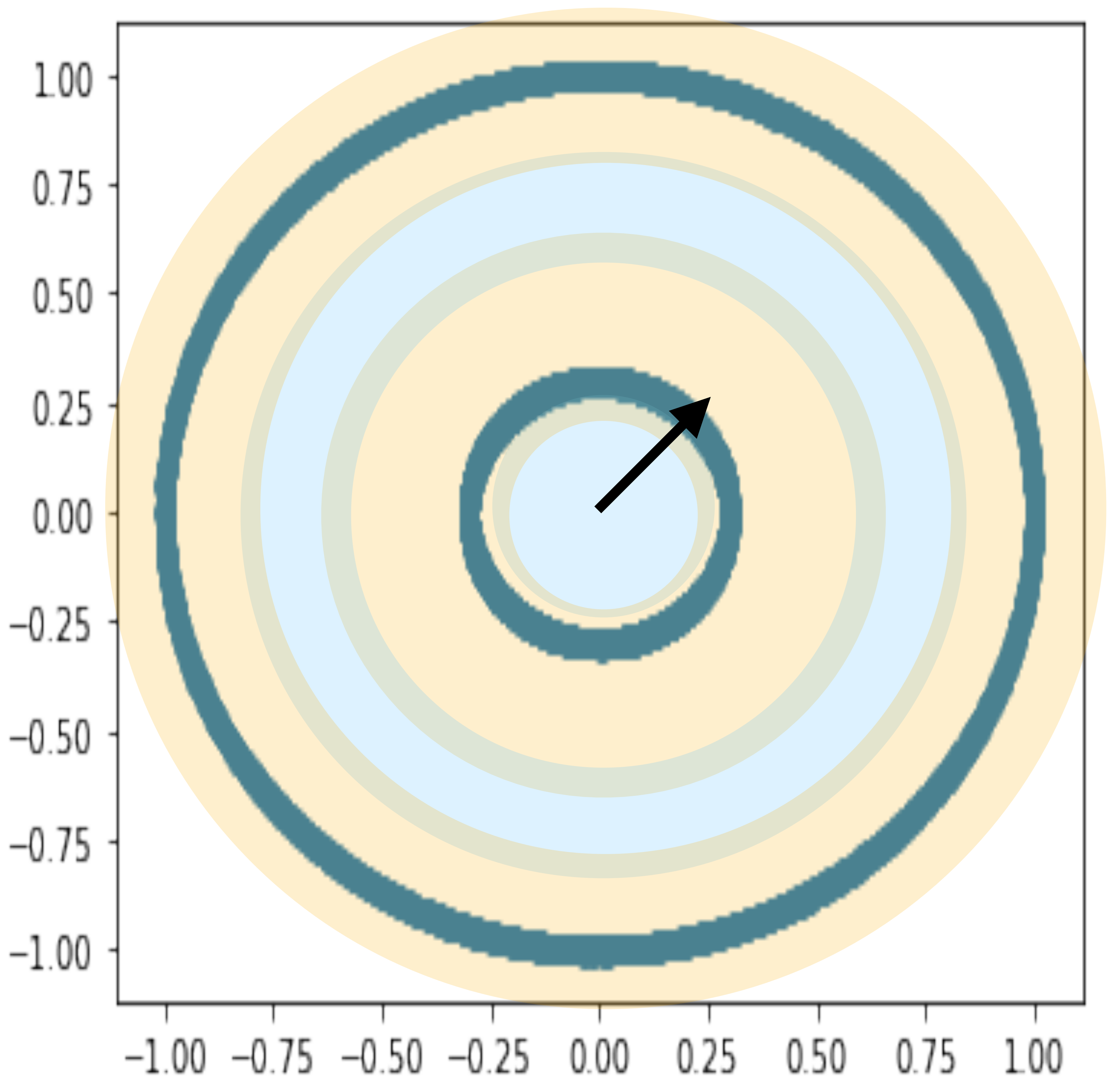


N = 10x10

Overlap = 0.6

Lens: x-y projection

Example 1: Two circles



N = 10

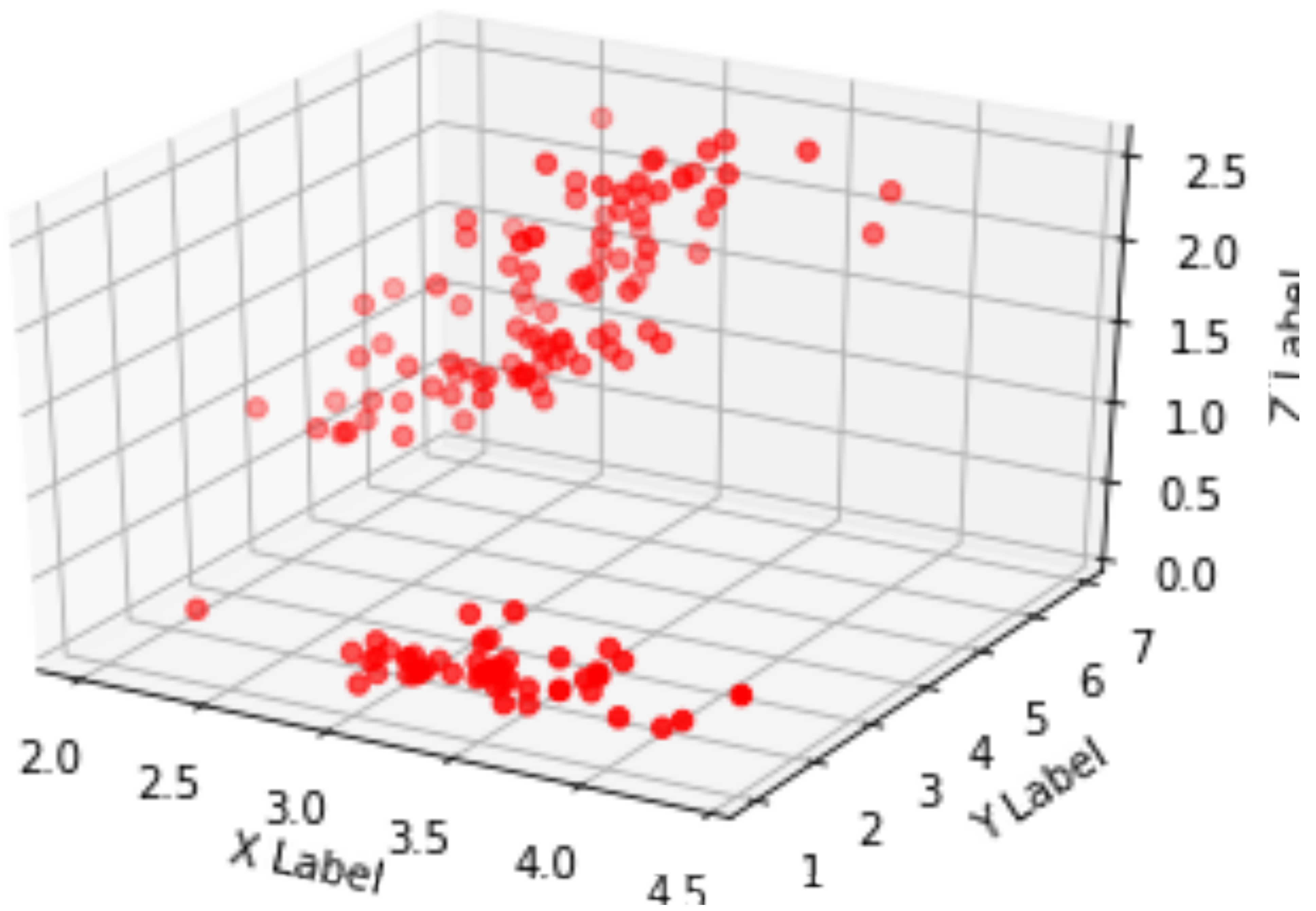
Overlap = 0.35

Lens: l2 norm

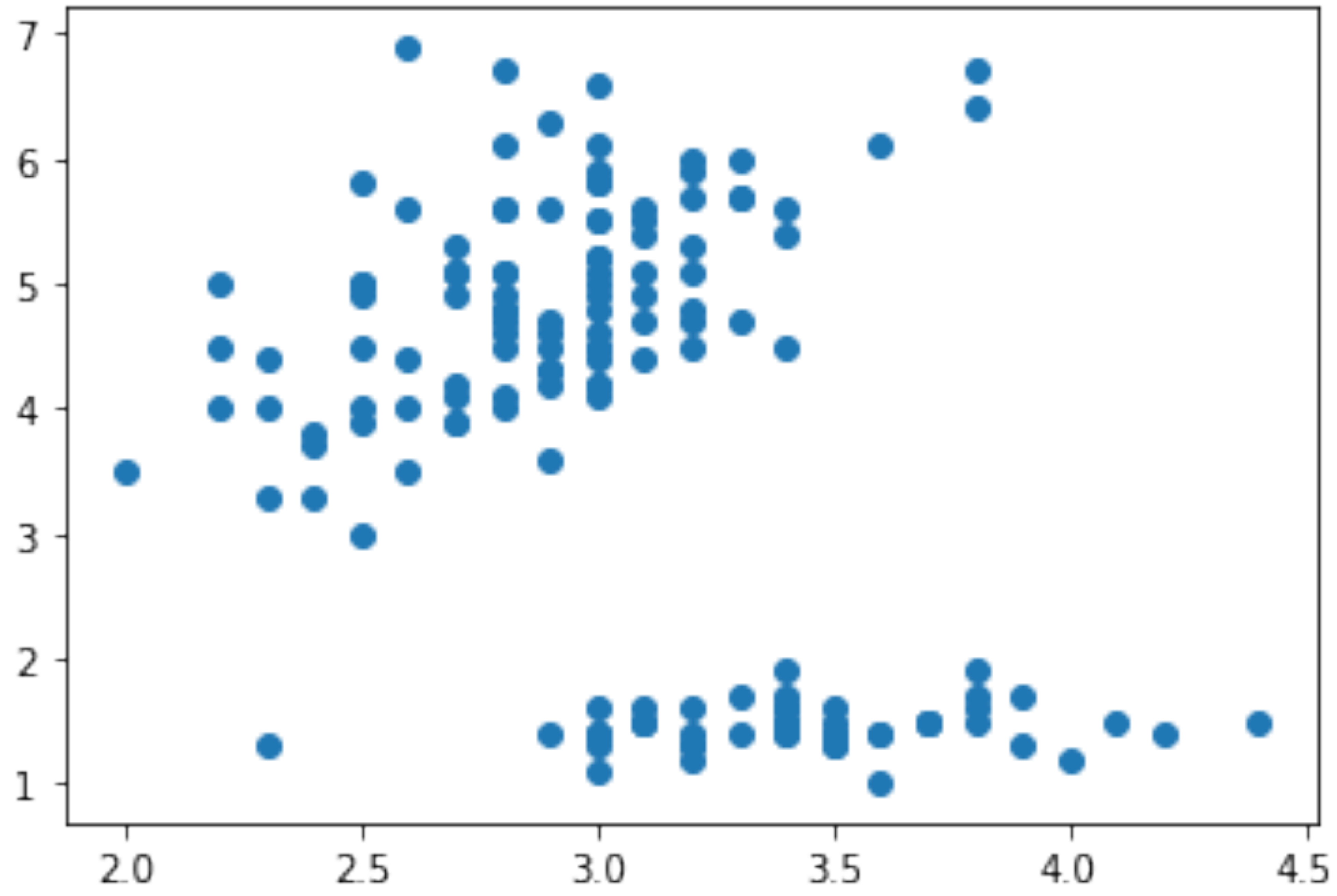
Example 2: Iris dataset

Iris dataset: 150 x 4 (150 points, 4 features, 3 types of flower)

3D plot 1-2-3 features



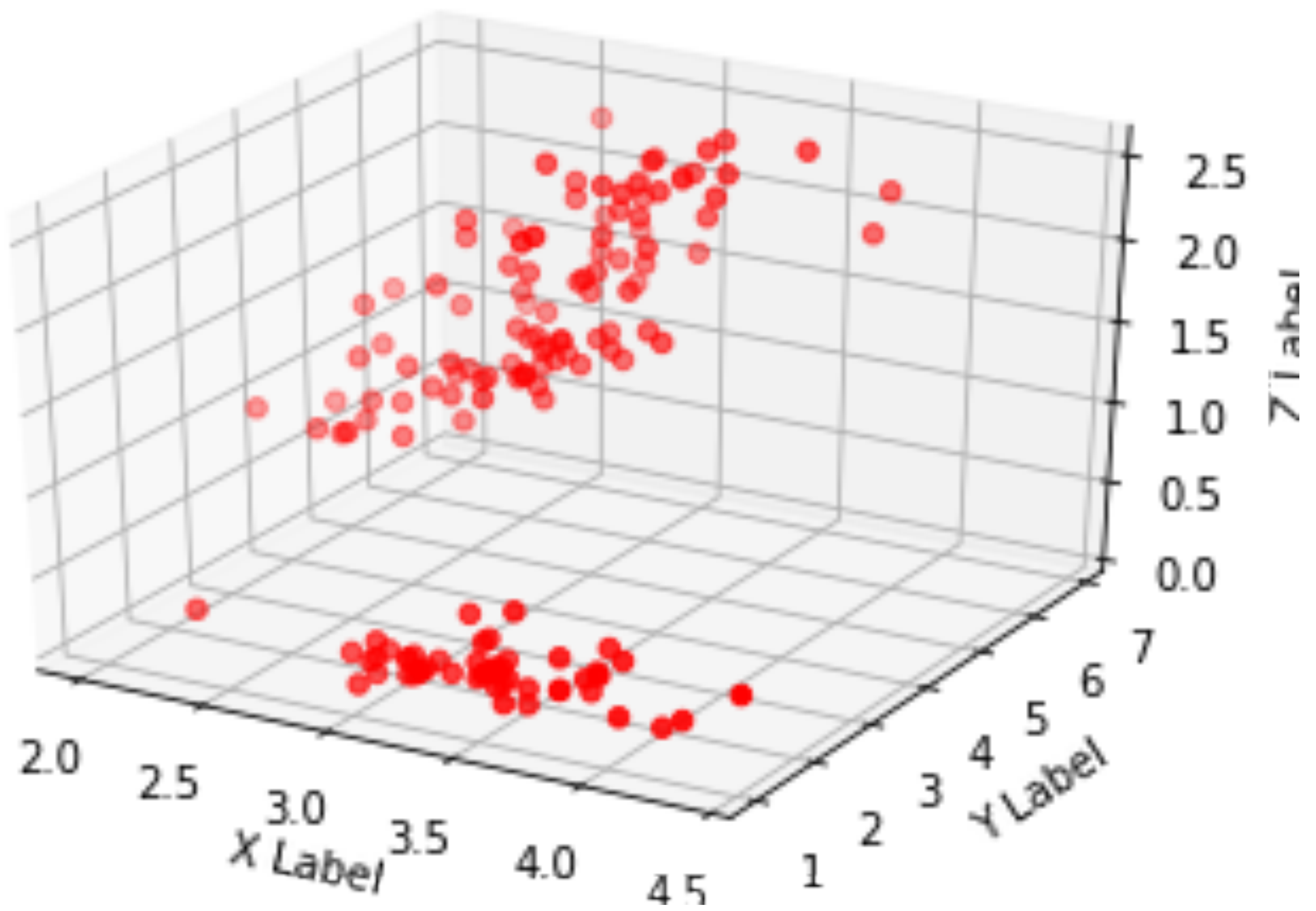
Lens: Projection to 1-2 plane



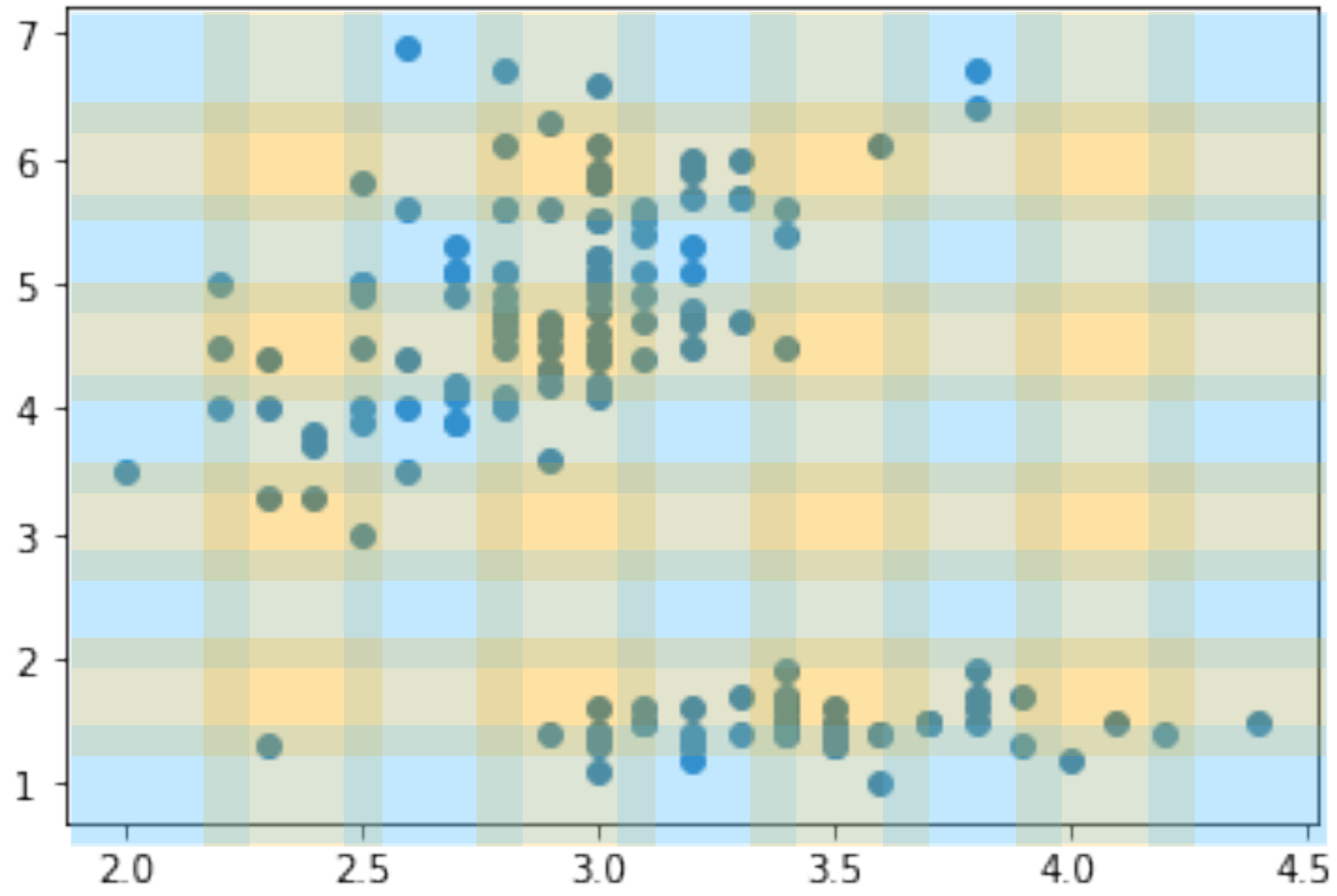
Example 2: Iris dataset

Iris dataset: 150 x 4 (150 points, 0-1-2-3 features, 3 types of flower)

3D plot 1-2-3 features



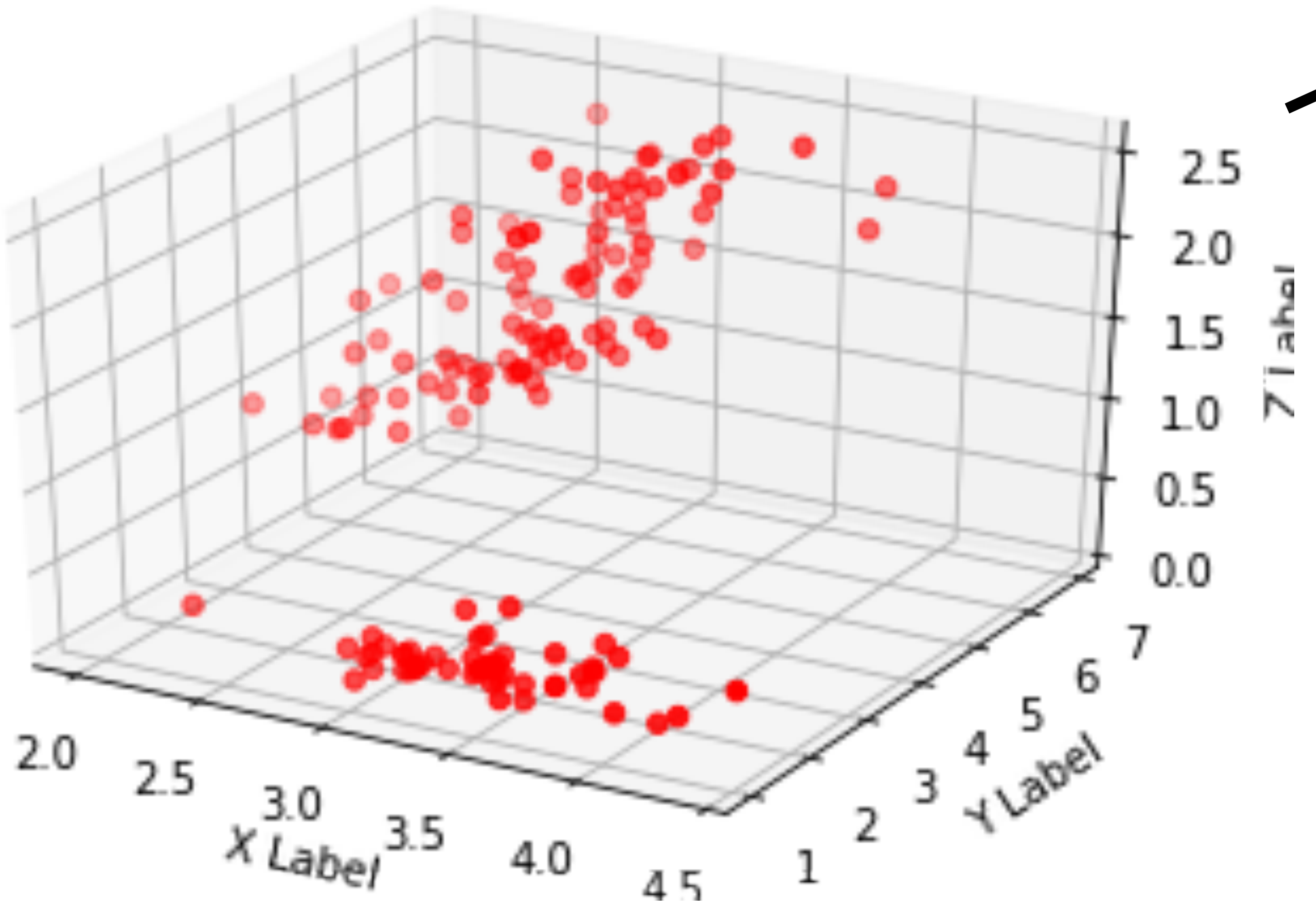
Lens: Projection to 1-2 plane

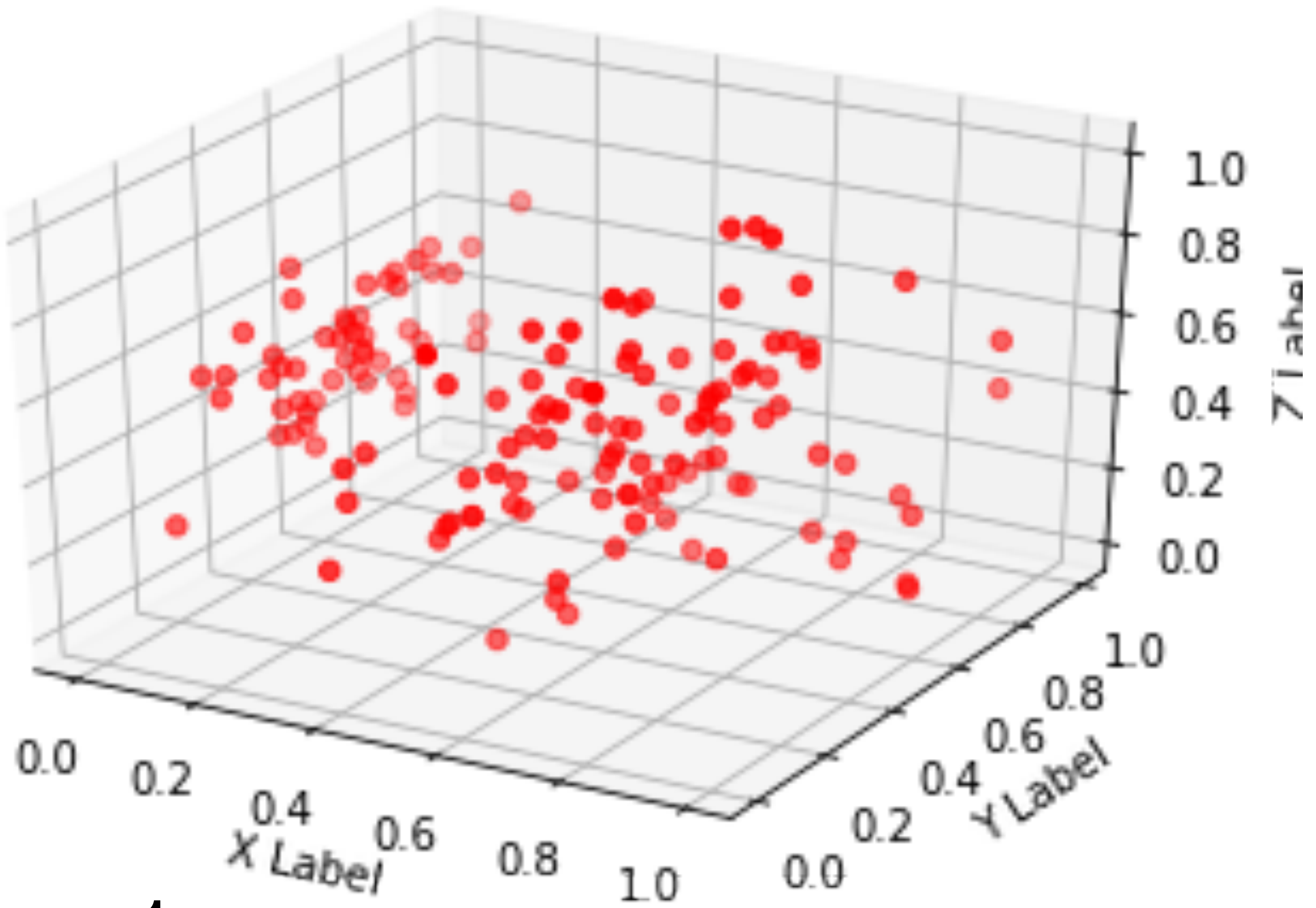


Example 2: Iris dataset

Concatenation of lens

Data (4D)



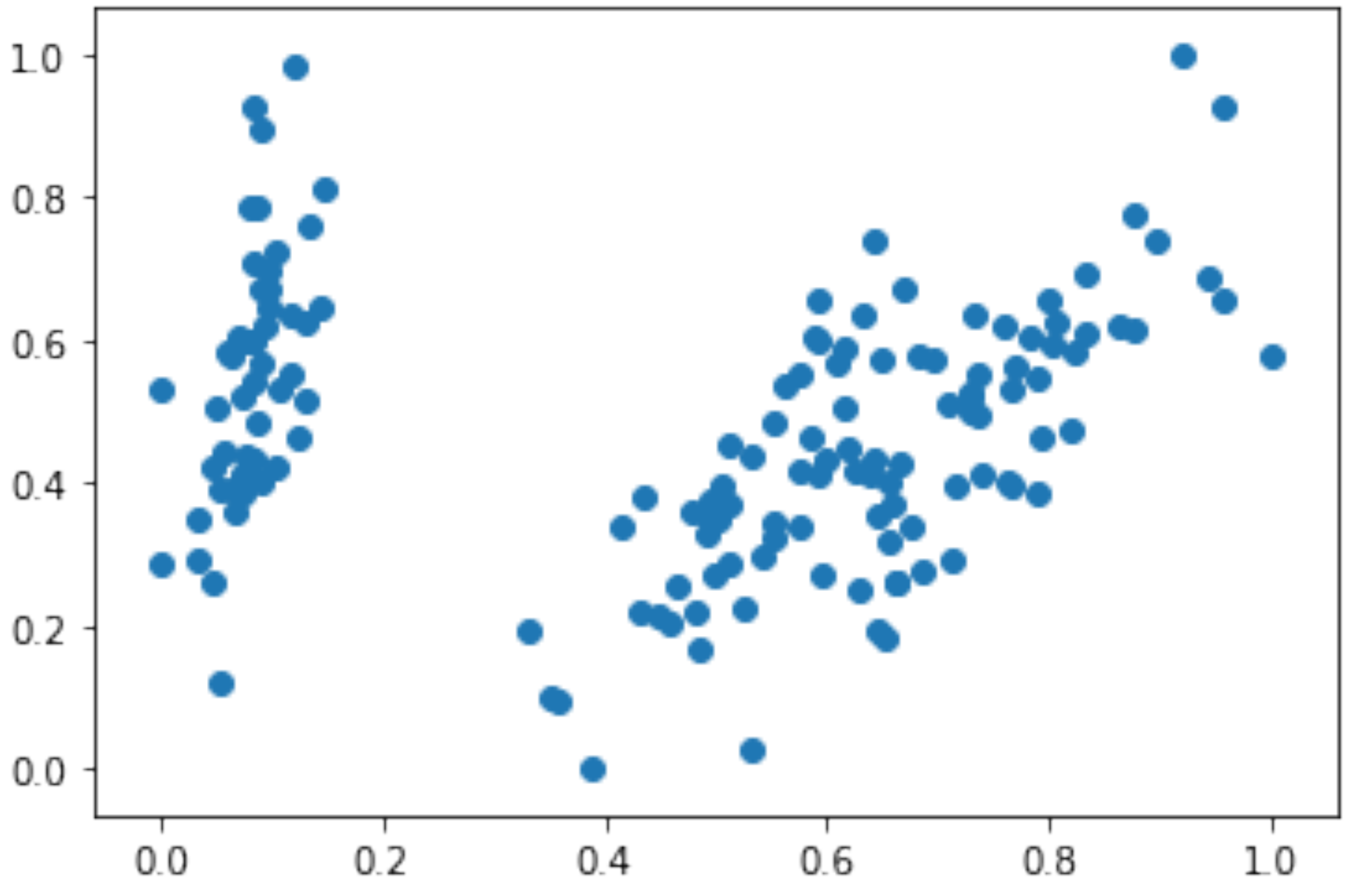


Lens1:

3D plot 3 comp. PCA

Lens 2:

Projection to 0-1 PCA comp.

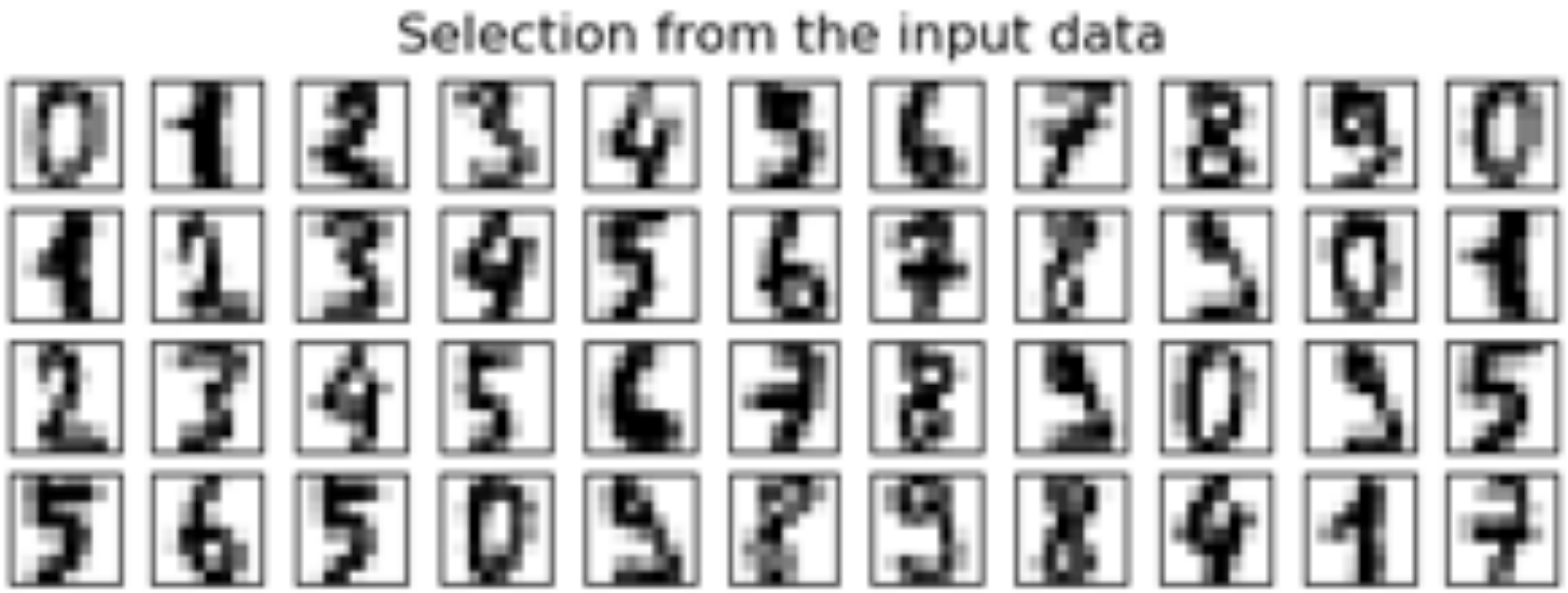


Trivial example, same as PCA 2 comp!

Numbers

t-SNE is a tool to visualize high-dimensional data. It converts similarities between data points to joint probabilities and tries to minimize the Kullback-Leibler divergence between the joint probabilities of the low-dimensional embedding and the high-dimensional data.

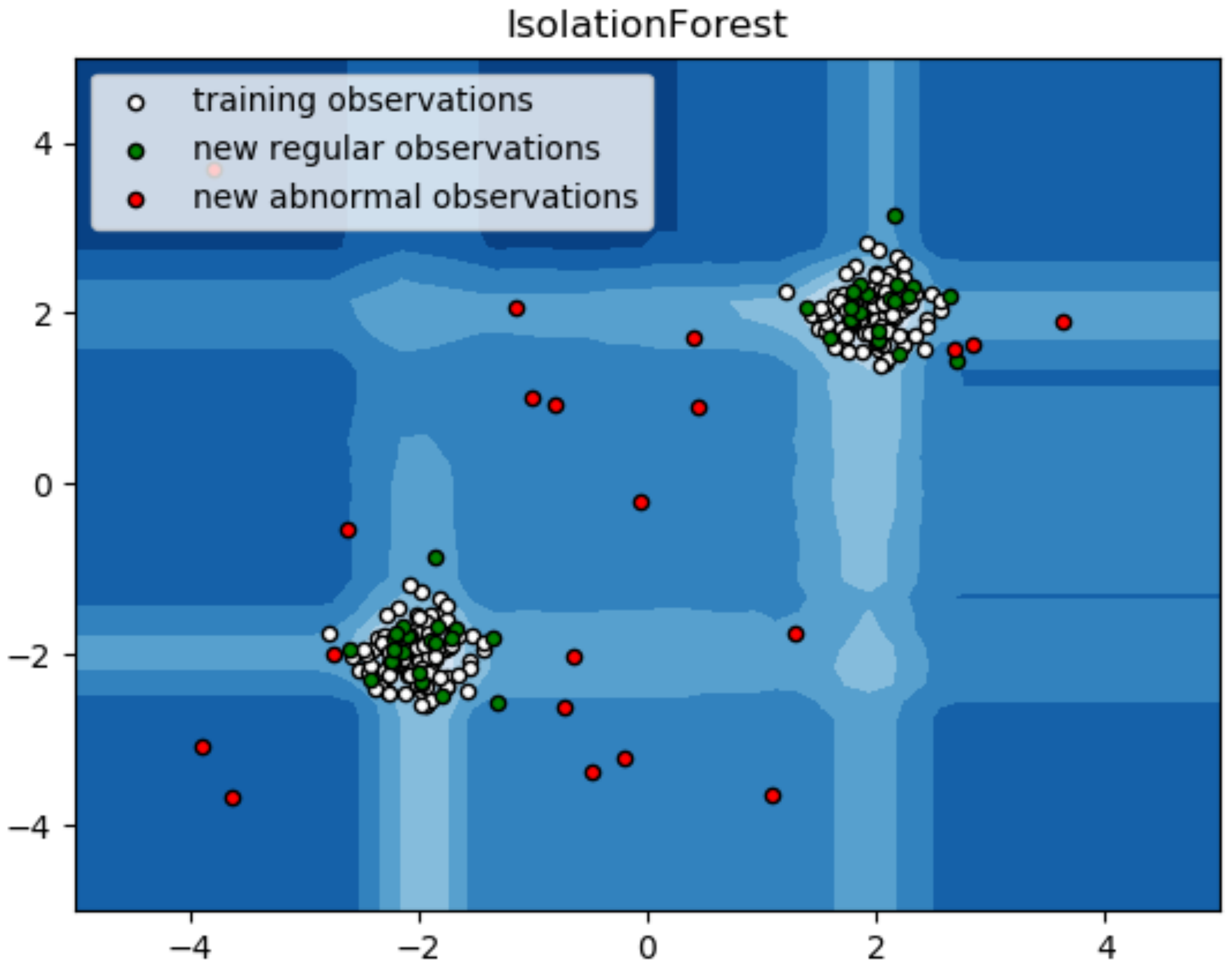
t-SNE has a cost function that is not convex, i.e. with different initializations we can get different results.



Breast cancer

The IsolationForest 'isolates' observations by randomly selecting a feature and then randomly selecting a split value between the maximum and minimum values of the selected feature.

Since recursive partitioning can be represented by a tree structure, the number of splittings required to isolate a sample is equivalent to the path length from the root node to the terminating node.



Breast cancer

Combine both lenses to create a 2-D [Isolation Forest, l2norm]

lens = np.c_[lens1, lens2] —> 2D lens

This time, instead of making a “lens chhain” we compute two 1D lenses

and combine together into a 2D lens

Links:

https://sauln.github.io/blog/mapper-intro/

https://www.youtube.com/watch?v=2PSqWBIrn90

https://kepler-mapper.scikit-tda.org/index.html

https://www.youtube.com/watch?v=h0bnG1Wavag