Mapper Graphs

Lecture 18 - CMSE 890

Prof. Elizabeth Munch

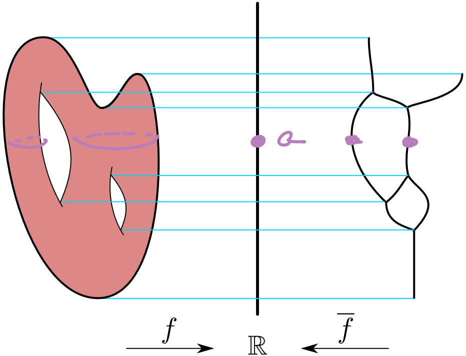
Michigan State University
::
Dept of Computational Mathematics, Science & Engineering

Tues, Nov 11, 2025

Goals for today

- 9.1, 9.3: Mapper Graphs

Recall: Reeb graph definition



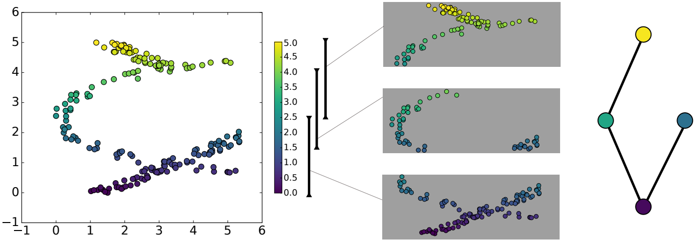
Given a function $f : X \to \mathbb{R}$.
Define an equivalence relation $\sim$ by $s \sim y$ iff

- $f(x) = f(y) = \alpha$
- x and y are in the same connected component of the level set $f^{-1}(\alpha)$.
- Let $[x]$ denote the equivalence class of $x \in X$.
- The Reeb graph R_f of $f : X \to \mathbb{R}$ is the quotient space $X/\sim$.
- Let $\Phi : X \to R_f$, $x \to [x]$ be the quotient map.

Section 1

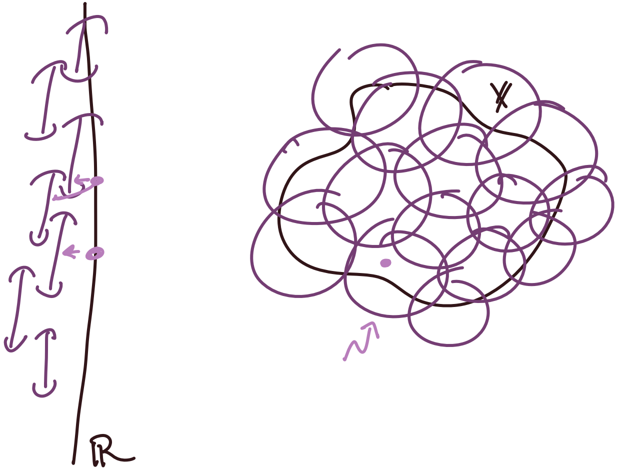
Mapper Graphs

Idea: Approximate Reeb graph when we have only a point cloud sample of the space.



Cover

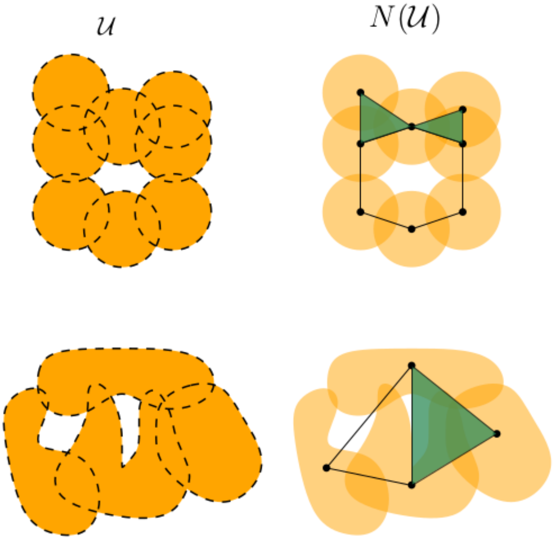
A cover of a set $\mathbb{X}$ is a collection of sets $\mathcal{U} = \{U_1, \cdots, U_k\}$ such that $\mathbb{X} \subseteq \bigcup_i U_i$.



Recall: Nerve

Given a finite collection of sets $\mathcal{F}$, the **nerve** is

$$\mathrm{Nrv}(\mathcal{U}) = \{X \subseteq \mathcal{F} \mid \bigcap_{U \in X} U \neq \emptyset\}.$$



Mapper graph paper

Eurographics Symposium on Point-Based Graphics (2007)
M. Botsch, R. Pajarola (Editors)

Topological Methods for the Analysis of High Dimensional Data Sets and 3D Object Recognition

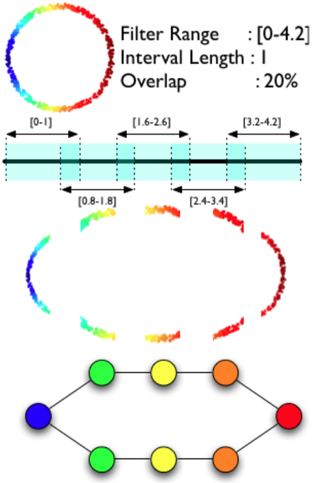
Gurjeet Singh[1] , Facundo Mémoli[2] and Gunnar Carlsson[†2]

[1]Institute for Computational and Mathematical Engineering, Stanford University, California, USA.
[2]Department of Mathematics, Stanford University, California, USA.

Abstract

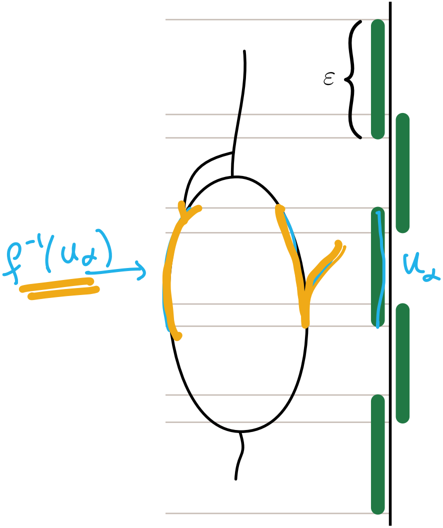
We present a computational method for extracting simple descriptions of high dimensional data sets in the form of simplicial complexes. Our method, called `Mapper`*, is based on the idea of* partial clustering *of the data guided by a set of functions defined on the data. The proposed method is not dependent on any particular clustering algorithm, i.e. any clustering algorithm may be used with* `Mapper`*. We implement this method and present a few sample applications in which simple descriptions of the data present important information about its structure.*

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modelling.



Mapper definition (continuous input version): Part I

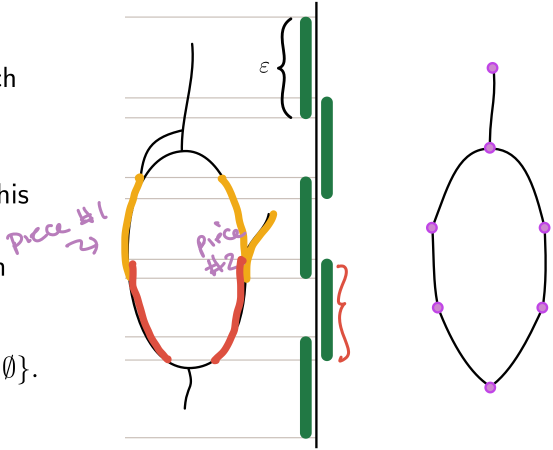
- Given $f : |K| \to \mathbb{R}$.
- Fix a cover $\mathcal{U} = \{U_\alpha\}$ of $\mathbb{R}$.
- The collection $f^{-1}(\mathcal{U}) = \{f^{-1}(U_\alpha)\}$ is a cover of $\mathbb{X}$.



Mapper definition (continuous input version): Part II

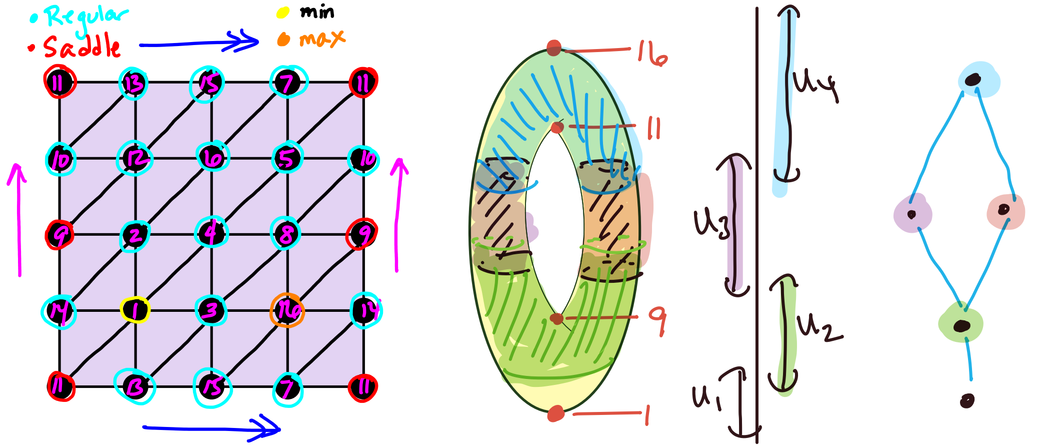
- Let $f^{-1}(\mathcal{U})^*$ be the cover which splits the sets into connected components.
- Then Mapper is the nerve of this cover.
  Recall: Given a finite collection of sets $\mathcal{F}$ in $\mathbb{R}^N$, the **nerve** is

$$\mathrm{Nrv}(\mathcal{F}) = \{X \subseteq \mathcal{F} \mid \bigcap_{U \in X} U \neq \emptyset\}.$$



What is the Mapper graph for the cover

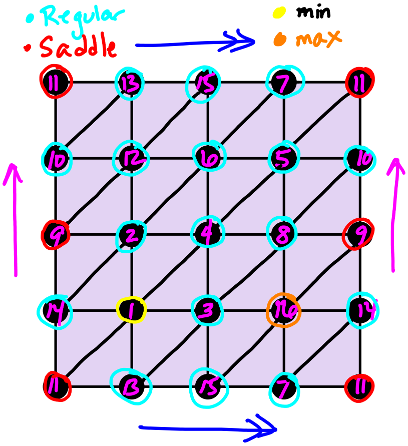
$$\{U_1 = (0,5), U_2 = (4,10), U_3 = (9.5,10.5), U_4 = (10,20)\}$$

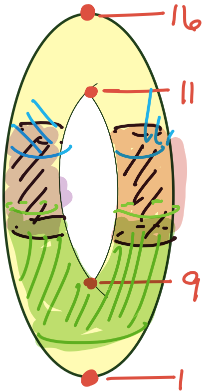


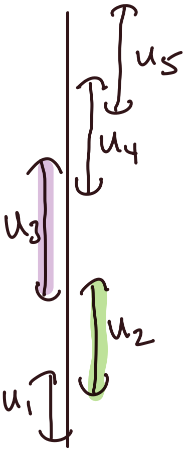
Try it:

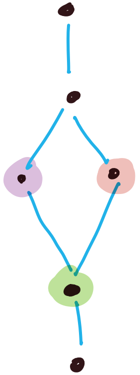
What is the Mapper graph for the cover

$$\{U_1 = (0,5), U_2 = (4,10), U_3 = (9.5,10.5), U_4 = \cancel{(10,20)}\}$$







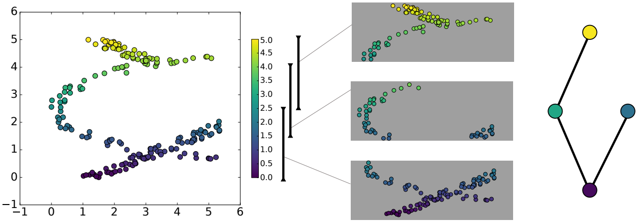


Section 2

Point cloud version

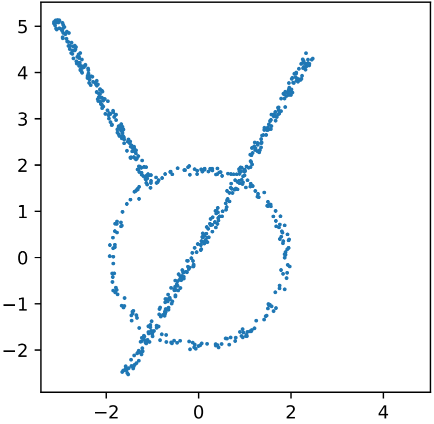
Point cloud mapper idea

1. Choose a $\mathbb{R}$-valued function on the data.
   - "Lens function"
2. Choose a cover of the range.
3. Cluster the points with values inside each cover element.
4. Construct the nerve.



Assumptions

Given point cloud P of N points with distance $d(x, y)$.



Next goal: What function if we don't already have one?

Lens Function: Density

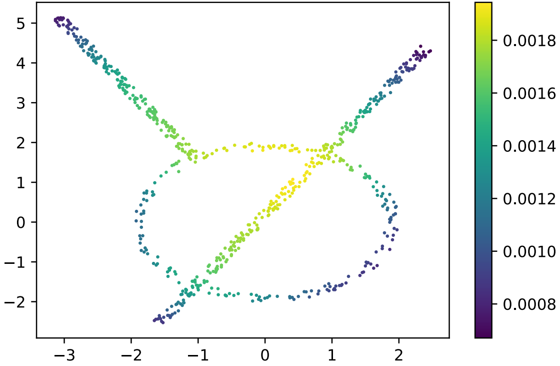
Idea: Choose function based on estimation of density.

Gaussian kernel version: For $x \in P$,

$$f_\varepsilon(x) = C_\varepsilon \sum_{y \in P} \exp\left(\frac{-d(x,y)^2}{\varepsilon}\right)$$

Parameters:

- $x, y \in P$
- $\varepsilon > 0$: Determines smoothness.
- C_ε is constant so that $\int f_\varepsilon(x)\, dx = 1$



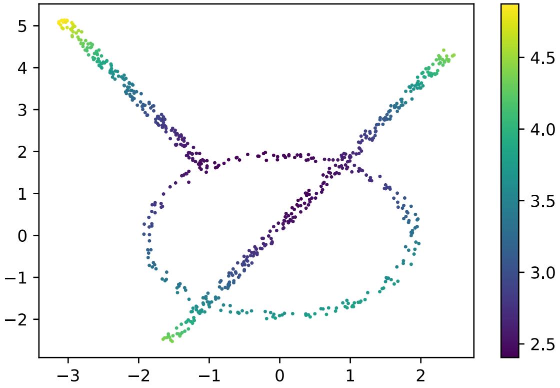
Density $f_{\cancel{3}}$ ε

Idea: low values correspond to points near the "center" of the data set, high correspond to points far from the center data set.

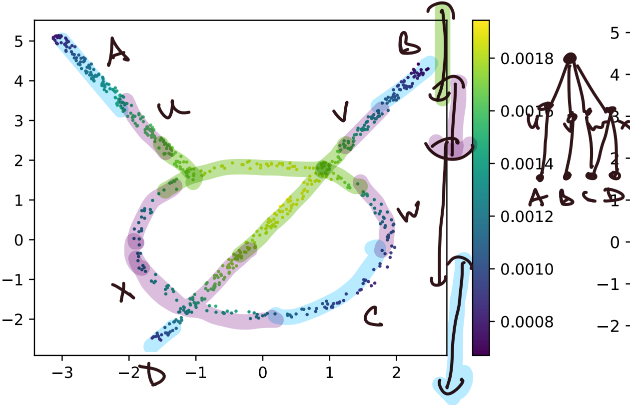
Given $1 \leq p < \infty$:

$$E_p(x) = \left(\frac{\sum_{y \in P} d(x,y)^p}{\underbrace{N}_{\#\text{ pts}}} \right)^{\frac{1}{p}}$$

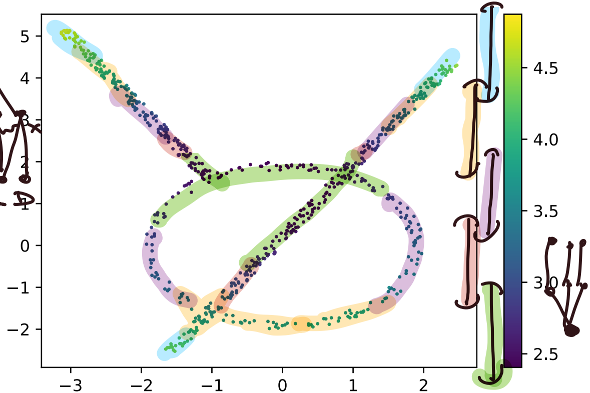
Note: No actual center has to be determined!



Eccentricity E_1

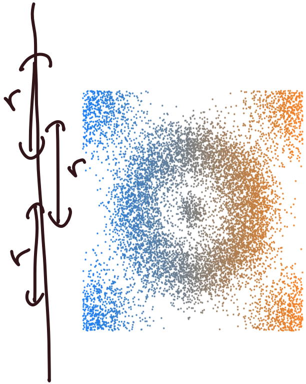


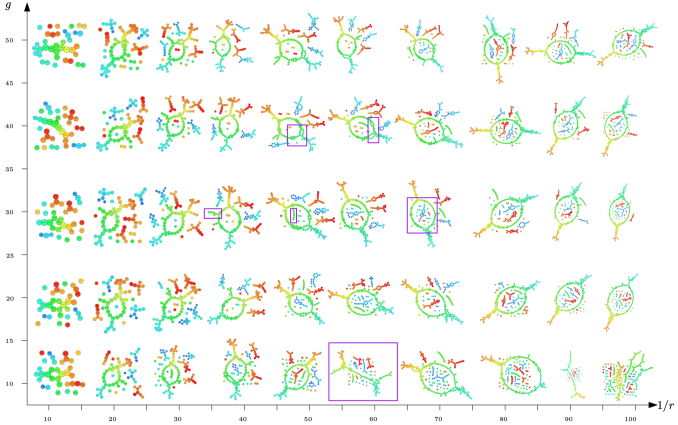
Density f_3



Eccentricity E_1

Cover choice is important!





larger intervals

smaller intervals

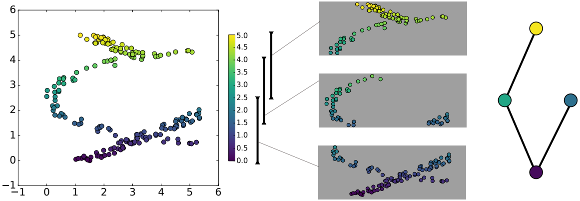
Interval length: r

Overlap: g

Image: Carriere et al. 2017

What do we want the clustering to do?

- Work on general metric spaces (not just point clouds in $\mathbb{R}^d$)
- Don't need to specify the number of clusters in advance

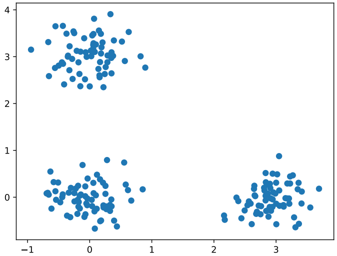


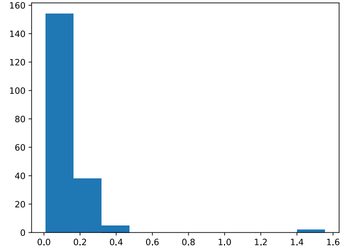
Choice of clustering: Histogram heuristic

Use the 0-dimensional persistence diagram

- All points are $(0, \mathrm{death})$.
- Draw the output as a histogram.
- Determine cutoff where there is an empty bin in the histogram.

Sometimes called *single-linkage clustering*





Choice of clustering: DBScan

- Doesn't require the number of clusters in advance.
- Based on density of points.

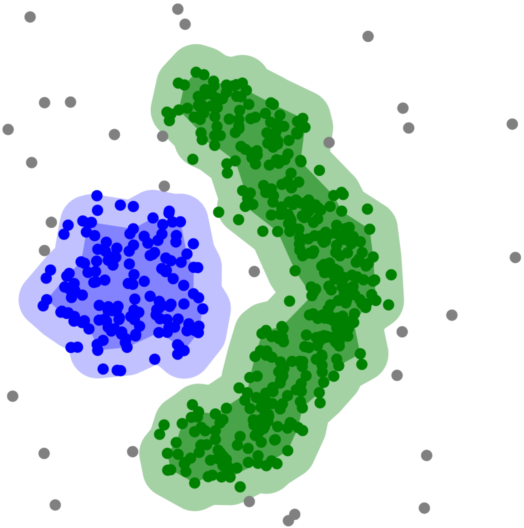
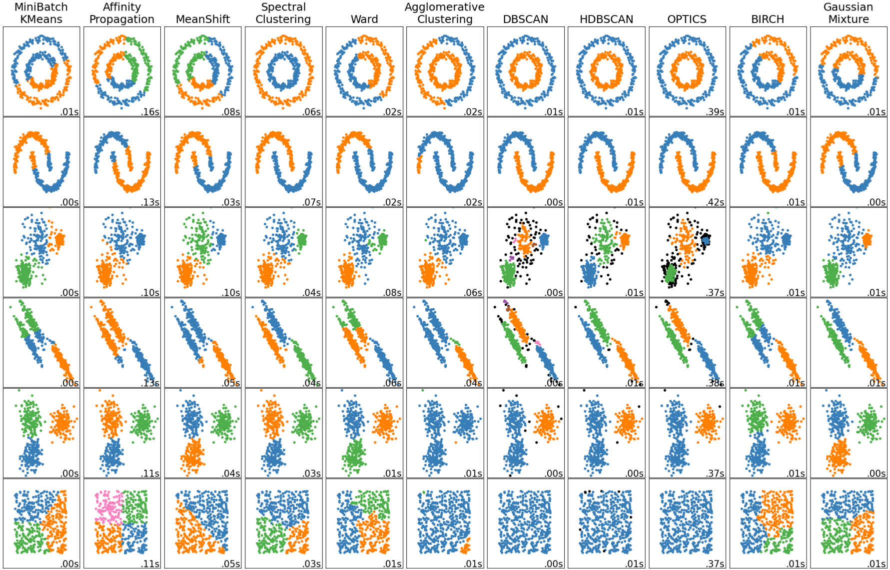
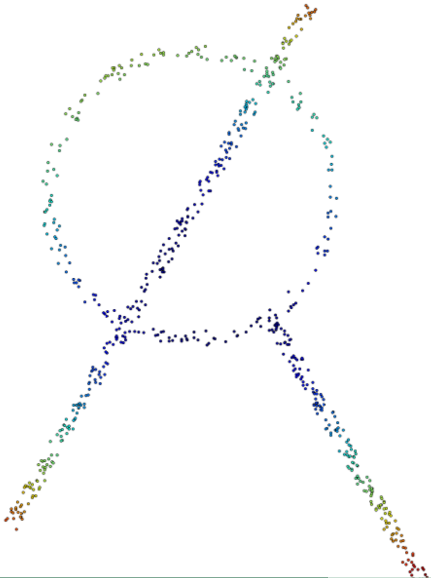


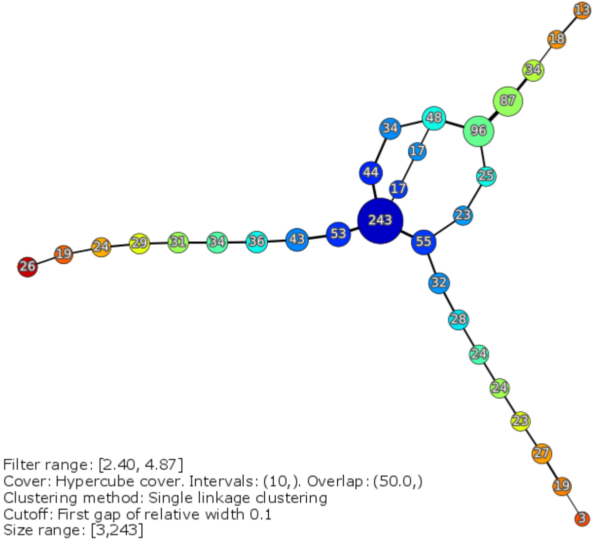
Image: Wikipedia

Clustering: More options



How it gets put back together





Notes on visualizations

- Size of nodes
- Choosing different colorings

Section 3

Breast Cancer Data

Breast Cancer Data Paper

PNAS

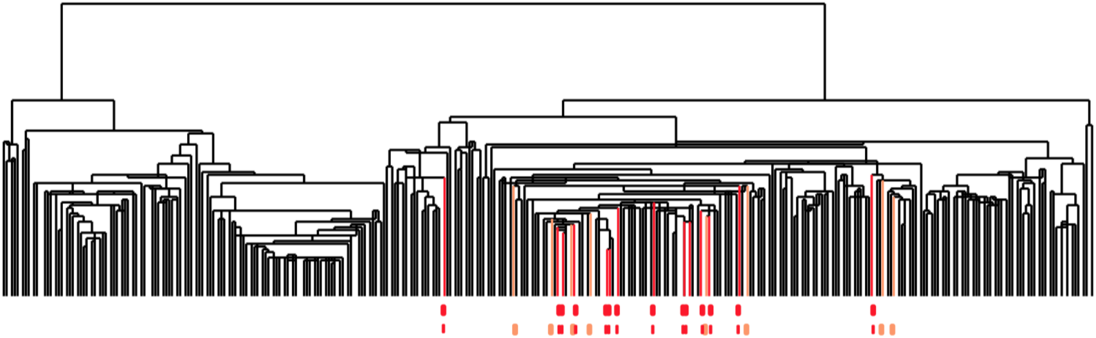
Topology based data analysis identifies a subgroup of breast cancers with a unique mutational profile and excellent survival

Monica Nicolau[a], Arnold J. Levine[b,1], and Gunnar Carlsson[a,c]

[a]Department of Mathematics, Stanford University, Stanford, CA 94305; [b]School of Natural Sciences, Institute for Advanced Study, Princeton, NJ 08540; and [c]Ayasdi, Inc., Palo Alto, CA 94301

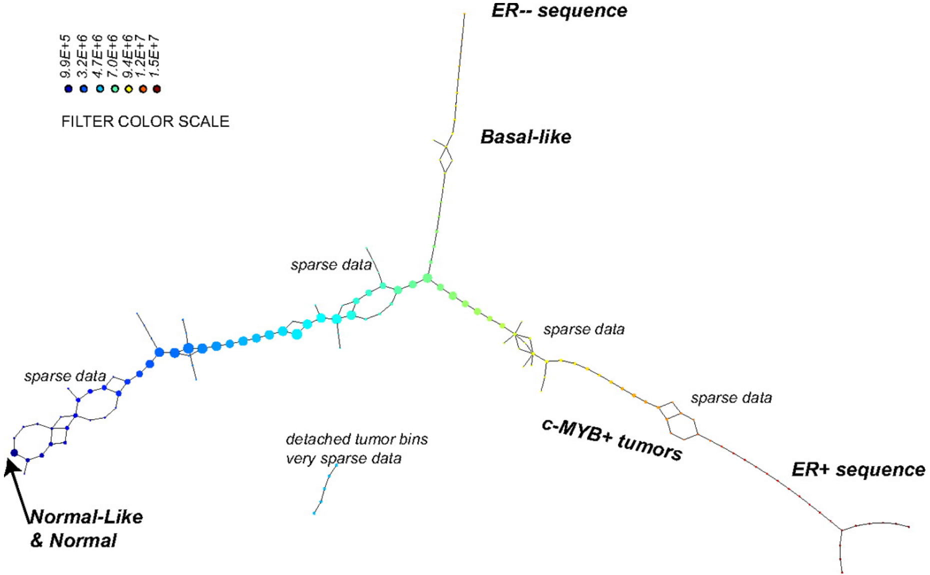
Contributed by Arnold J. Levine, February 25, 2011 (sent for review July 23, 2010)

Clustering of data



Red: Patients with poor survival outcome (c-MYB+ tumors)

The Mapper Graph



Section 4

Ball Mapper

Ball mapper paper

Ball mapper: a shape summary for topological data analysis.

Paweł Dłotko
Department of Mathematics, Swansea University

January 23, 2019

Abstract

Topological data analysis provides a collection of tools to encapsulate and summarize the shape of data. Currently it is mainly restricted to *mapper algorithm* and *persistent homology*. In this paper we introduce new mapper–inspired descriptor that can be applied for exploratory data analysis.

arXiv: 1901.07410

Algorithm 1 Greedy ε-net

Input: Point cloud X, $\varepsilon > 0$

Mark all points in X as *not covered*.

Create initially empty cover vector $B(X, \varepsilon)$.

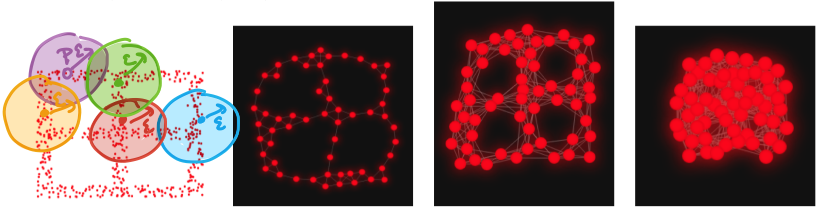
repeat

  Pick a first point $p \in X$ that is not covered.

  For every point in $x \in B(p, \varepsilon) \cap X$, add p to $B(X, \varepsilon)[x]$.

until All elements of X are covered.

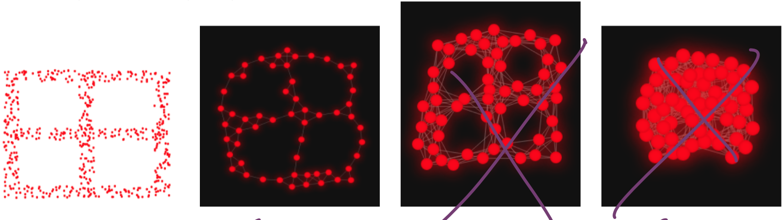
return $B(X, \varepsilon)$.



Ball mapper - Definition

Algorithm 3 Construction of Ball Mapper graph

Input: Cover vector $B(X, \varepsilon)$ from Algorithm $\boxed{1}$ or $\boxed{2}$.
V = cover elements in $B(X, \varepsilon)$, $E = \emptyset$,
for Every point $p \in X$ **do**
  For every pair of cover elements $c_1 \neq c_2$ in $B(X, \varepsilon)[p]$, add a (weighted) edge between vertices corresponding to the cover elements c_1 and c_2. Formally, $E = E \cup \{c_1, c_2\}$
Return $G = (V, E)$

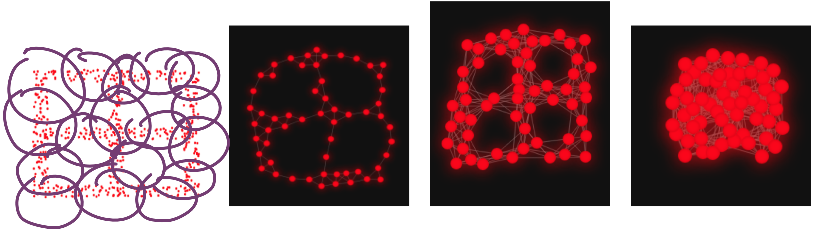


$\varepsilon = 1$ $\varepsilon = 2$ $\varepsilon = 3$

Multiscale Ball Mapper

Algorithm 4 Multi-scale Ball Mapper algorithm.

Input: Point cloud X, $\varepsilon_1 \leq \varepsilon_2 \leq \ldots \leq \varepsilon_n$.
Pick a collection B of ball centers using Algorithm 1 or Algorithm 2 for X and ε_1.
for Every $\varepsilon \in \{\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n\}$ **do**
  For every point $p \in X$, select those elements in B which are closer than ε to p and, based on them, construct the cover vector $B(X, \varepsilon)$.
  Compute graph G_ε by running Algorithm 3 for $B(X, \varepsilon)$.
Return $\{G_{\varepsilon_1}, \ldots, G_{\varepsilon_n}\}$



Is it mapper?

Similarities

Nerve construction

Still results in graph
(for nice covers)

Differences

No function to
start with

Next time

- Code
- Bring laptop!