Differentiable Mapper

Topological Optimization of Data Representation

Ziyad Oulhaj [1] Mathieu Carrière [2] Bertrand Michel [1]

[1]Laboratoire de Mathématiques Jean Leray
Nantes Université - École Centrale Nantes

[2]DataShape - Centre Inria d'Université Côte d'Azur

May 10, 2024

Contents

Background on Mapper Graphs

Reeb Graph

Let X be a topological space and let $f : X \to \mathbb{R}$ be a continuous function called *filter function*. Let $\sim_f$ be the equivalence relation between two elements x and y in X defined by: $x \sim_f y$ if and only if x and y are in the same connected component of $f^{-1}(z)$ for some z in $f(X)$.
The Reeb graph $\mathrm{R}_f(X)$ of X is then simply defined as the quotient space $X/ \sim_f$.

Reeb Graph example

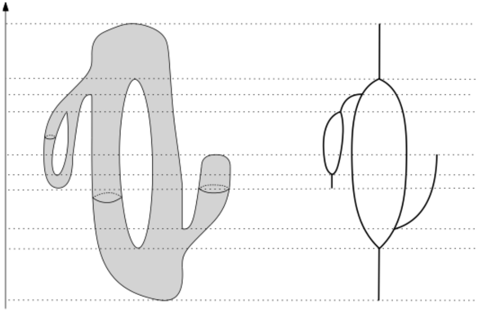


Figure: Example of a Reeb graph (of a double torus) taken from [Carriere et al., 2018].

Mapper Algorithm

The Mapper was introduced in [Singh et al., 2007] as a discrete version of the Reeb graph $\mathrm{R}_f(\mathcal{X})$.
Given a discrete metric space $(\mathbb{X}_n = \{X_1, \ldots, X_n\}, d)$, as well as a filter function f :

1. Cover the range of values $\mathbb{Y}_n = f(\mathbb{X}_n)$ with a set of consecutive intervals $I_1, \ldots, I_r$ that overlap, i.e., one has $I_i \cap I_{i+1} \neq \varnothing$ for all $1 \leq i \leq r-1$.
2. Apply a clustering algorithm to each pre-image $f^{-1}(I_j)$, $j \in \{1, ..., r\}$. This defines a *pullback cover* $\mathcal{C} = \{\mathcal{C}_{1,1}, \ldots, \mathcal{C}_{1,k_1}, \ldots, \mathcal{C}_{r,1}, \ldots, \mathcal{C}_{r,k_r}\}$ of $\mathbb{X}_n$.
3. The Mapper graph is defined as the *nerve* of $\mathcal{C}$. Each node $v_{j,k}$ of the Mapper graph corresponds to an element $\mathcal{C}_{j,k}$ of $\mathcal{C}$, and two nodes $v_{j,k}$ and $v_{j',k'}$ are connected by an edge if and only if $\mathcal{C}_{j,k} \cap \mathcal{C}_{j',k'} \neq \varnothing$.

Mapper Example

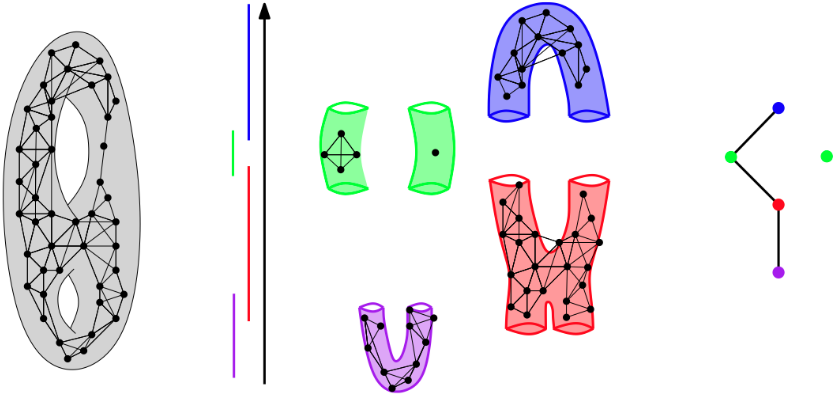


Figure: Example of a Mapper graph taken from [Carriere et al., 2018]. The clustering is specified from a neighborhood graph.

Hyperparameters

Most commonly, the intervals are chosen to have the same length, with a fixed percentage of overlap between consecutive ones.

Hyperparameters to choose

1. Number of intervals (resolution) : r,
2. Percentage of overlap (gain) : g,
3. Clustering,
4. Filter function.

Hyperparameters

Most commonly, the intervals are chosen to have the same length, with a fixed percentage of overlap between consecutive ones.

Hyperparameters to choose

1. Number of intervals (resolution) : r, $\leftarrow$ Grid search, Look for stability, Infer it from the inherent manifold structure of the data.
2. Percentage of overlap (gain) : g, $\leftarrow$ Same as resolution.
3. Clustering, $\leftarrow$ Depends on the nature of the dataset.
4. Filter function. $\leftarrow$?

Topological Signatures on Mapper graphs

Filtering a Mapper graph

Fix a dataset (discrete metric space) $\mathbb{X}_n$, a integer r and a clustering function Clus.
Let $\mathbb{K}$ be the set of simplicial complexes of dimension less or equal to 1 (i.e., graphs) and such that their sets of vertices (i.e., their $0-$skeletons) are subsets of the power set $\mathcal{P}(\mathbb{X}_n)$.

Filtering a Mapper graph

Fix a dataset (discrete metric space) $\mathbb{X}_n$, a integer r and a clustering function Clus.
Let $\mathbb{K}$ be the set of simplicial complexes of dimension less or equal to 1 (i.e., graphs) and such that their sets of vertices (i.e., their $0-$skeletons) are subsets of the power set $\mathcal{P}(\mathbb{X}_n)$.

Filtration values

For a function $F \in \mathcal{F}(\mathbb{X}_n, \mathbb{R})$, we associate a filtration ϕ to some $K \in \mathbb{K}$ with:

$$\forall \sigma \in K \ : \ \phi(\sigma) = \max_{c \in \sigma} \frac{\sum_{x \in c} F(x)}{\mathrm{card}(c)}.$$

Filtering a Mapper graph

Fix a dataset (discrete metric space) $\mathbb{X}_n$, a integer r and a clustering function Clus.
Let $\mathbb{K}$ be the set of simplicial complexes of dimension less or equal to 1 (i.e., graphs) and such that their sets of vertices (i.e., their $0-$skeletons) are subsets of the power set $\mathcal{P}(\mathbb{X}_n)$.

Filtration values

For a function $F \in \mathcal{F}(\mathbb{X}_n, \mathbb{R})$, we associate a filtration ϕ to some $K \in \mathbb{K}$ with:

$$\forall \sigma \in K \ : \ \phi(\sigma) = \max_{c \in \sigma} \frac{\sum_{x \in c} F(x)}{\mathrm{card}(c)}.$$

Persistence diagram

We denote by MapPers the function that takes a Mapper graph and a scalar function on $\mathbb{X}_n$, and then outputs the persistence diagram:

$$\mathrm{MapPers}\colon \mathbb{K} \times \mathcal{F}(\mathbb{X}_n, \mathbb{R}) \longrightarrow \mathcal{P}(\mathbb{R}^2).$$

Persistence loss

Denoting PD as the set of subsets of $\mathbb{R}^2$ consisting of a finite number of points outside the diagonal $\Delta = \{(x,x) : x \in \mathbb{R}\}$, we also consider a loss function : $\ell \colon PD \longrightarrow \mathbb{R}$.
For example :

$$\{(u_i, v_i)\}_{1 \le i \le n} \longmapsto -\sum_{i=1}^{n} |u_i - v_i|.$$

Soft Mapper

Formal definition of the Mapper

Cover assignment

We call any binary matrix $e \in \{0,1\}^{n \times r}$ an *r-latent cover assignment* of $\mathbb{X}_n$, where $e_{i,j} = 1$ must be understood as point x_i belonging to the j-th element of a *latent cover* of the data.

Formal definition of the Mapper

Cover assignment

We call any binary matrix $e \in \{0,1\}^{n \times r}$ an *r-latent cover assignment* of $\mathbb{X}_n$, where $e_{i,j} = 1$ must be understood as point x_i belonging to the j-th element of a *latent cover* of the data.

Mapper complex generating function

Define

$$\mathrm{MapComp}\colon \{0,1\}^{n \times r} \longrightarrow \mathbb{K},$$

to be the function that takes a cover assignment and associates its corresponding Mapper complex. It uses the same algorithm for Mapper but replaces $f^{-1}(I_j)$ by $\{x_i \,:\, e_{i,j} = 1\}$.

Soft Mapper

We use this formalism to define distributions on Mapper graphs.

Cover assignment scheme

A *cover assignment scheme* is a double indexed sequence of random variables

$$A = (A_{i,j})_{\substack{1 \le i \le n \\ 1 \le j \le r}}$$

such that each $A_{i,j}$ is a Bernoulli random variable conditionally to $\mathbb{X}_n$.
Let $p_{i,j}(\mathbb{X}_n)$ be the parameter of the Bernoulli distribution of $(A_{i,j} | \mathbb{X}_n)$, which is thus a function of the point cloud $\mathbb{X}_n$.

Soft Mapper

We use this formalism to define distributions on Mapper graphs.

Cover assignment scheme

A *cover assignment scheme* is a double indexed sequence of random variables

$$A = (A_{i,j})_{\substack{1 \leq i \leq n \\ 1 \leq j \leq r}}$$

such that each $A_{i,j}$ is a Bernoulli random variable conditionally to $\mathbb{X}_n$.
Let $p_{i,j}(\mathbb{X}_n)$ be the parameter of the Bernoulli distribution of $(A_{i,j}|\mathbb{X}_n)$, which is thus a function of the point cloud $\mathbb{X}_n$.

Soft Mapper

Let A be a cover assignment scheme. The *Soft Mapper* of A is defined as the associated distribution of Mapper complexes, which corresponds to the push forward measure of the distribution of A by the map MapComp.

Example 1 : The regular case

let $f : \mathbb{X}_n \to \mathbb{R}$ be a filter function and let $(I_j)_{1 \le j \le r}$ be a finite cover of the image $f(\mathbb{X}_n)$ of f. The standard Mapper graph is then defined as $\mathrm{MapComp}(e^*)$, where for every $1 \le i \le n$ and $1 \le j \le r$:

$$e^*_{i,j} = 1 \text{ if } f(x_i) \in I_j.$$

The cover assignment scheme A^*, in this case, is degenerate at e^*.

$$\mathbb{P}(A^* = e|\mathbb{X}_n) = \begin{cases} 1 & \text{if } e = e^*, \\ 0 & \text{otherwise.} \end{cases}$$

Example 2 : Smooth relaxation of the regular case

Let $\delta > 0$. Using the same notations as before, and denoting each element of the cover with $I_j = [a_j, b_j]$, consider, for each $j \in \{1, ..., r\}$, the function $q_j : \mathbb{X}_n \longrightarrow [0,1]$ defined with:

$$x \mapsto \begin{cases} 1, & \text{if } f(x) \in [a_j, b_j] \\ \exp(1 - 1/(1 - (\frac{a_j - f(x)}{\delta})^2)), & \text{if } f(x) \in [a_j - \delta, a_j] \\ \exp(1 - 1/(1 - (\frac{f(x) - b_j}{\delta})^2)), & \text{if } f(x) \in [b_j, b_j + \delta] \\ 0, & \text{otherwise} \end{cases}$$

Now, define $A_\delta = (A_{\delta,i,j})_{\substack{1 \leq i \leq n \\ 1 \leq j \leq r}}$ to be the random variable in $\{0,1\}^{n \times r}$ such that for every $(i,j) \in \{1, ..., n\} \times \{1, ..., r\}$:

$$A_{\delta,i,j} \mid \mathbb{X}_n \sim \mathcal{B}(q_j(x_i)).$$

We have

$$A_\delta \xrightarrow[\delta \to 0]{\mathcal{L}} A^*.$$

Comparison between A^* and A_δ

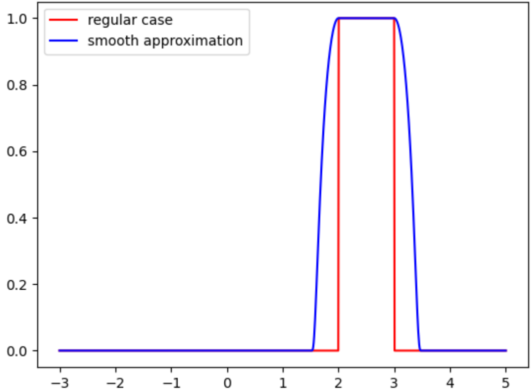
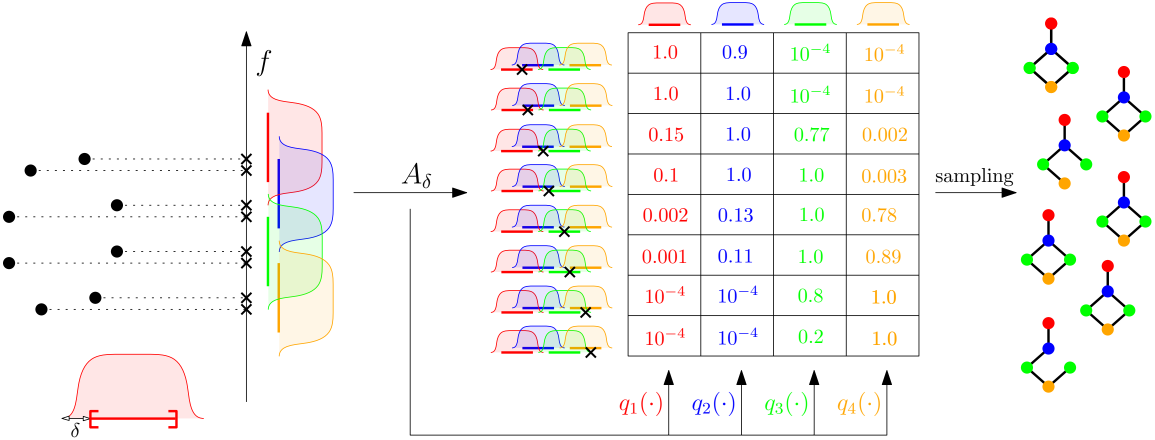


Figure: Assignment probability $p_{i,j}(\mathbb{X}_n)$ of a point x_i to an interval $I_j = [2, 3]$ for A^* and A_δ, plotted against $f(x_i)$.

Illustration of the smooth assignment scheme A_δ



1.0	0.9	10^{-4}	10^{-4}
1.0	1.0	10^{-4}	10^{-4}
0.15	1.0	0.77	0.002
0.1	1.0	1.0	0.003
0.002	0.13	1.0	0.78
0.001	0.11	1.0	0.89
10^{-4}	10^{-4}	0.8	1.0
10^{-4}	10^{-4}	0.2	1.0

Risk of a Soft Mapper

We define the risk of a Soft Mapper MapComp(A) by integrating the loss according to the distribution of the Soft Mapper :

$$\mathbb{E}\left(\mathcal{L}(A, F)|\mathbb{X}_n\right) = \sum_{e \in \{0,1\}^{n \times r}} \mathcal{L}(e, F) \cdot \mathbb{P}(A = e|\mathbb{X}_n).$$

Filter Optimization

Problem setting

Let us introduce a parameterized family of functions $\{f_\theta : \mathbb{X}_n \to \mathbb{R},\ \theta \in \mathbb{R}^s\}$. Let A be a cover assignment scheme whose joint distribution $\mathbb{P}_\theta$ depends on the filter function f_θ. Denoting

$$
\begin{aligned}
\mathrm{L} \colon \mathbb{R}^s &\longrightarrow \mathbb{R} \\
\theta &\longmapsto \mathbb{E}_\theta(\mathcal{L}(A, f_\theta)|\mathbb{X}_n),
\end{aligned}
\tag{1}
$$

our aim is to find a minimizer of L.

Guarantees on stochastic gradient descent

[Davis et al., 2020]

Under technical conditions on the stochastic gradient descent algorithm and under the following assumptions :

- L is definable in an o-minimal structure,
- L is locally Lipschitz,

then $(L(x_k))_k$ converges almost surely to a critical value and the limit points of $(x_k)_k$ are critical points of L.

Main result

[Oulhaj et al., 2024]

Suppose that there exists an o-minimal structure $\mathcal{S}$ such that:

- for every $x \in \mathbb{X}_n$, the function $\theta \mapsto f_\theta(x)$ is definable in $\mathcal{S}$ and is locally Lipschitz,
- for every $m \in \mathbb{N}$, the restriction of ℓ to the set of (extended) persistence diagrams of size m is definable in $\mathcal{S}$ and is locally Lipschitz,
- for every $e \in \{0, 1\}^{n \times r}$, the function $\theta \mapsto \mathbb{P}_\theta(A = e | \mathbb{X}_n)$ is definable in $\mathcal{S}$ and is locally Lipschitz.

Then L is definable in $\mathcal{S}$ and is locally Lipschitz.

Main result

[Oulhaj et al., 2024]

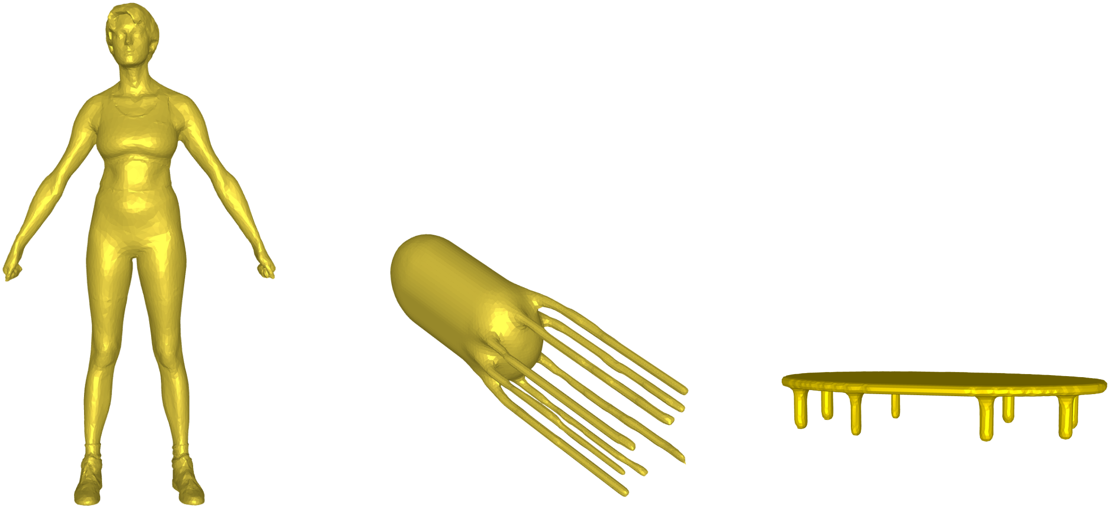
Suppose that there exists an o-minimal structure $\mathcal{S}$ such that:

- for every $x \in \mathbb{X}_n$, the function $\theta \mapsto f_\theta(x)$ is definable in $\mathcal{S}$ and is locally Lipschitz,
- for every $m \in \mathbb{N}$, the restriction of ℓ to the set of (extended) persistence diagrams of size m is definable in $\mathcal{S}$ and is locally Lipschitz,
- for every $e \in \{0,1\}^{n \times r}$, the function $\theta \mapsto \mathbb{P}_\theta(A = e|\mathbb{X}_n)$ is definable in $\mathcal{S}$ and is locally Lipschitz. $\leftarrow$ Doesn't work in the regular (degenerate) case

Then L is definable in $\mathcal{S}$ and is locally Lipschitz.

Experiments

3D shapes



We wish to optimize a linear parametric family of functions, i.e., equal to $\{f_\theta \colon x \mapsto \langle x, \theta \rangle,\ \theta \in \mathbb{R}^3\}$, and the cover assignment scheme A_δ is the smooth relaxation of the standard case, with $\delta = 10^{-2} \cdot (\max_{x \in \mathbb{X}_n} f_\theta(x) - \min_{x \in \mathbb{X}_n} f_\theta(x))$.

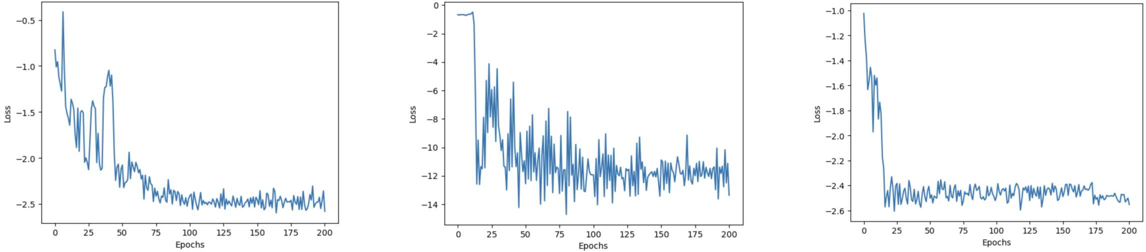
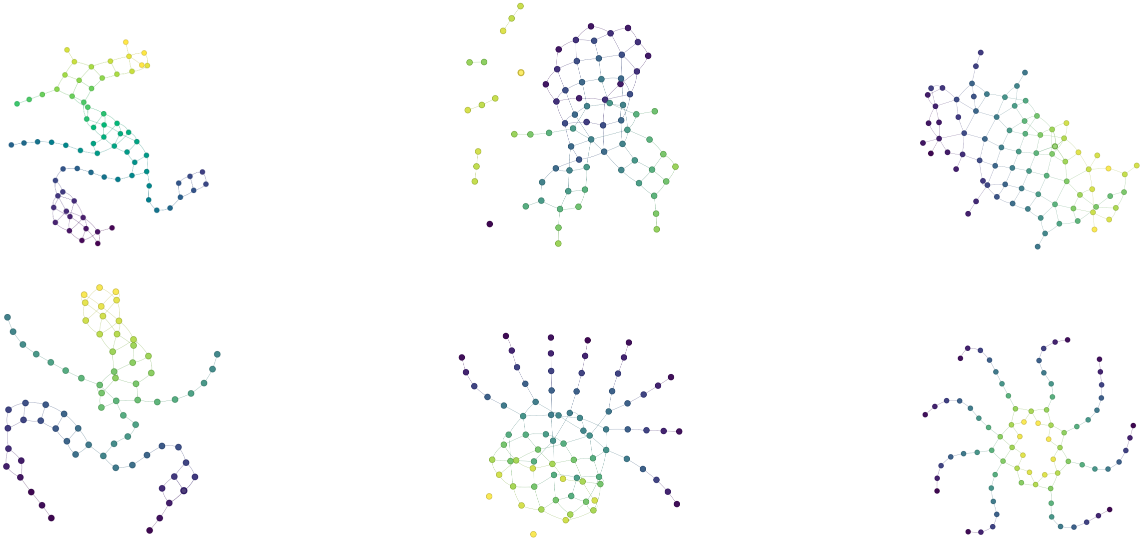


Figure: Learning curves for the 3-dimensional shapes corresponding, from left to right, to: the human, the octopus and the table.

Before - After



Human preimplantaion cells

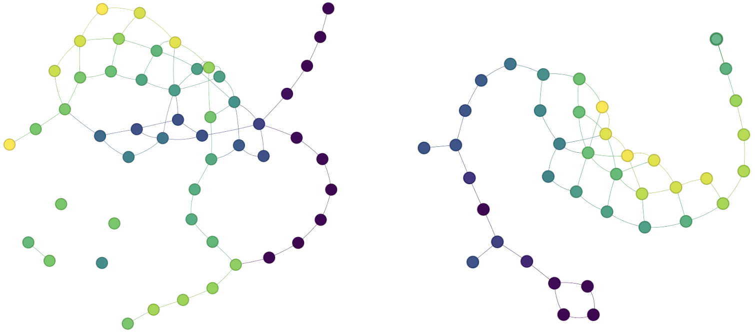
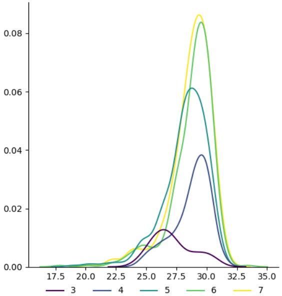


Figure: Regular Mapper graphs for the human preimplantation dataset computed using: in the left the initial filter function and in the right the optimized filter function. Vertices are colored using the mean value of the sampling timepoint in the clusters.

Correlation with the sampling time



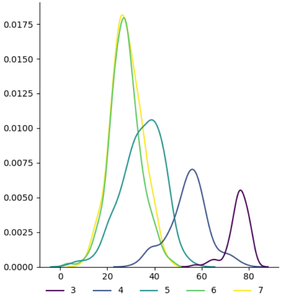


Figure: Estimated density of each subset of cells having the same sampling timepoint, with respect to: in the left the initial filter function values and in the right the optimized filter function values. Colors indicate the sampling timepoint in days.

Gene expressions

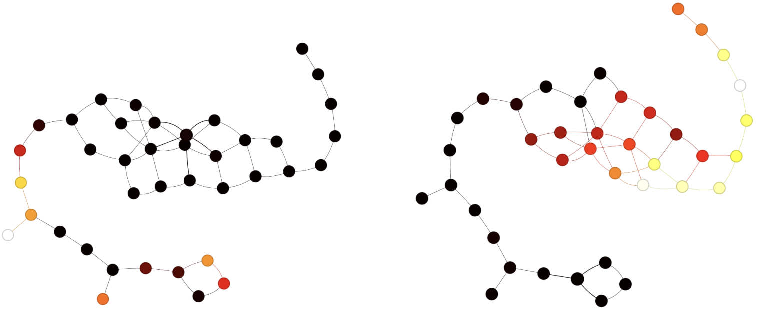


Figure: Regular Mapper graph computed using the optimized filter function, colored using the mean normalized expression of: in the left gene HTR3E and in the right gene CDX1.

Thank you for listening !

Carriere, M., Michel, B., and Oudot, S. (2018).
Statistical analysis and parameter selection for mapper.
The Journal of Machine Learning Research, 19(1):478–516.

Davis, D., Drusvyatskiy, D., Kakade, S., and Lee, J. D. (2020).
Stochastic subgradient method converges on tame functions.
Foundations of computational mathematics, 20(1):119–154.

Oulhaj, Z., Carrière, M., and Michel, B. (2024).
Differentiable mapper for topological optimization of data representation.
arXiv preprint arXiv:2402.12854.

Singh, G., Mémoli, F., Carlsson, G. E., et al. (2007).
Topological methods for the analysis of high dimensional data sets and 3d object recognition.
PBG@ Eurographics, 2:091–100.