Plan of the course

1. ToMATo for colocalizing cell types

   [Bae et al. - 2022 - *STopover captures spatial colocalization and interaction in the tumor microenvironment using topological analysis in spatial transcriptomics data*]

2. Rips persistence for marker gene correlations

   [Alsaleh et al. - 2022 - *Spatial transcriptomic analysis reveals associations between genes and cellular topology in breast and prostate cancers*]

3. Multi-persistence for immune cell arrangements

   [Vipond et al. - 2021 - *Multiparameter persistent homology landscapes identify immune cell spatial patterns in tumors*]

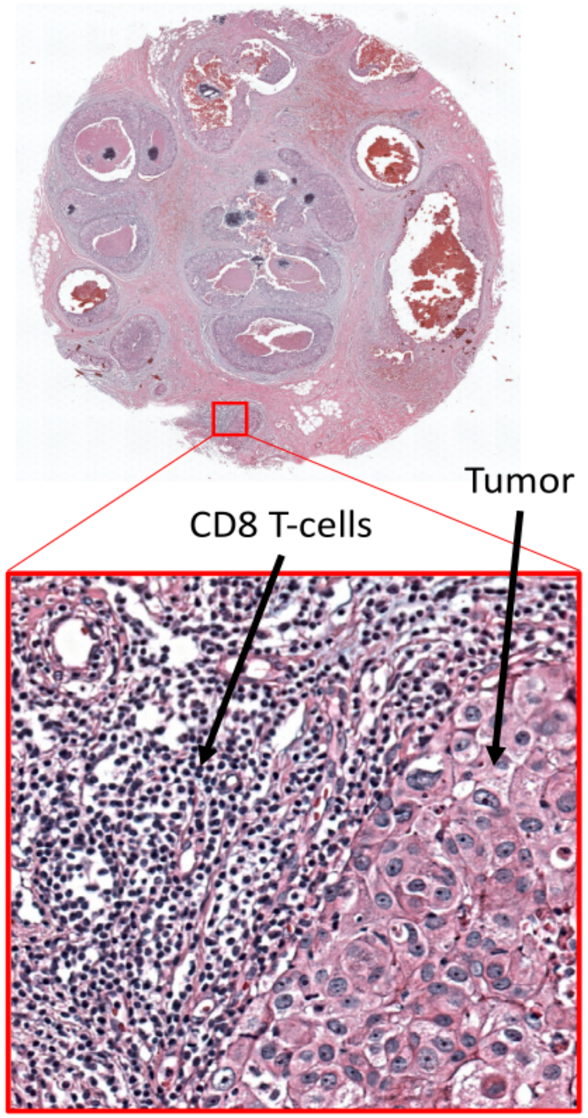
   [Benjamin et al. - 2022 - *Multiscale topology classifies and quantifies cell types in subcellular spatial transcriptomics*]

4. Future research directions

2. Rips persistence for marker gene correlations

2. Rips persistence for marker gene correlations

Q: What are the relations and correlations between *spatial features* and *marker genes*? Can one influence the other?

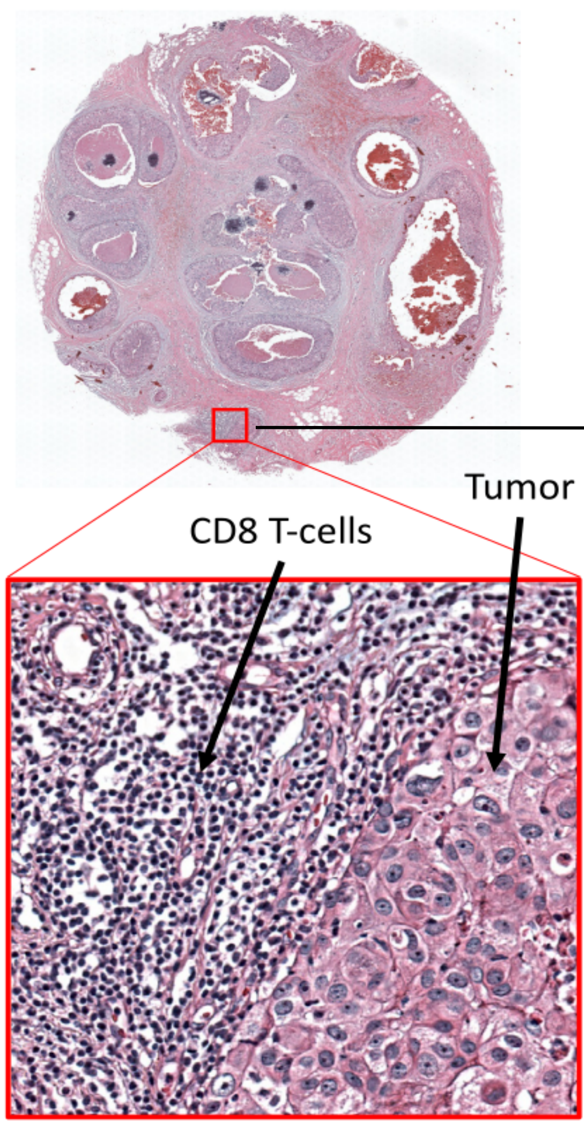


Local point cloud

$$P = \{p_1, \ldots, p_m\} \subset \mathbb{R}^2$$

2. Rips persistence for marker gene correlations

Q: What are the relations and correlations between *spatial features* and *marker genes*? Can one influence the other?



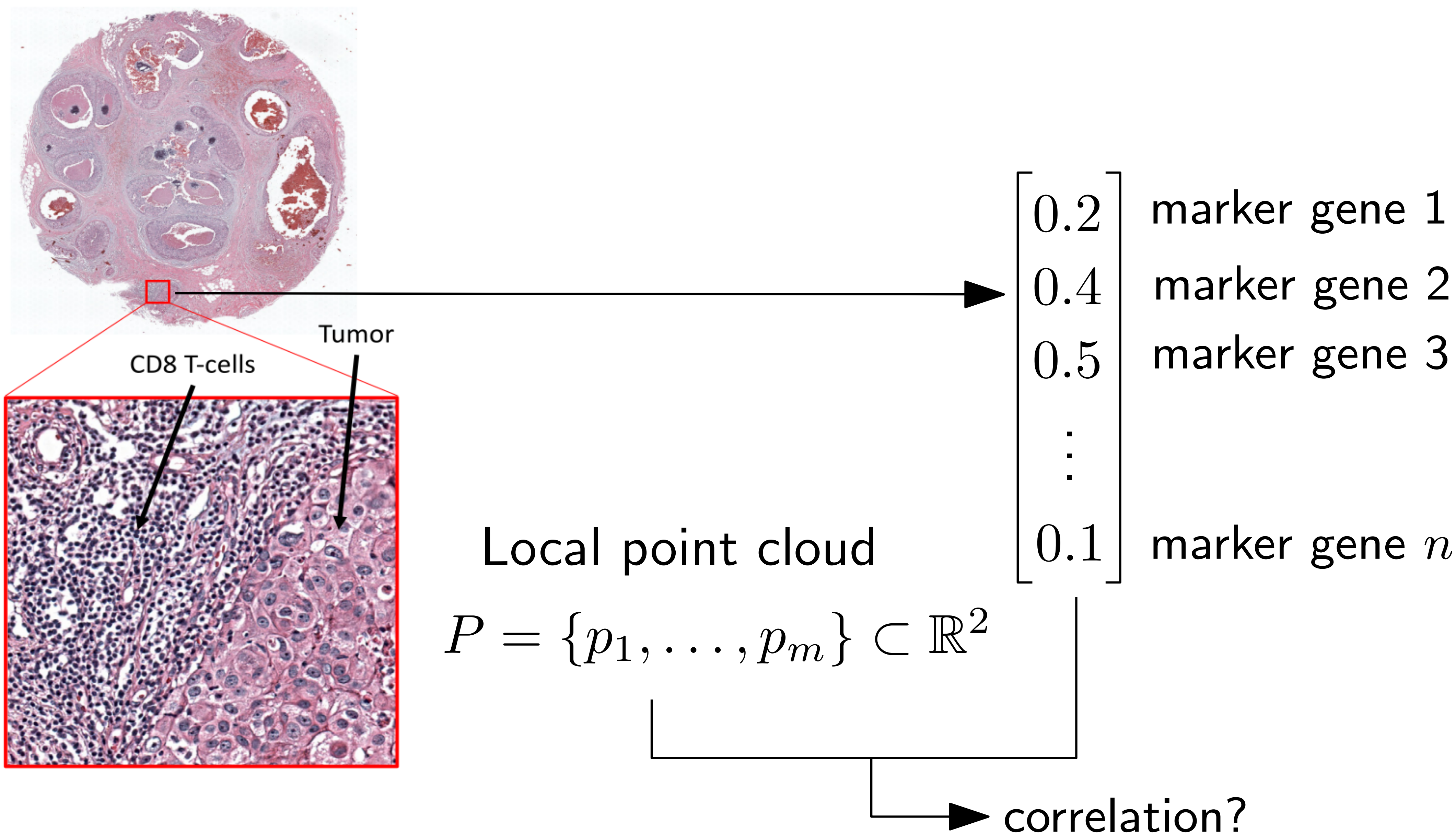
$$\begin{bmatrix} 0.2 \\ 0.4 \\ 0.5 \\ \vdots \\ 0.1 \end{bmatrix} \begin{matrix} \text{marker gene } 1 \\ \text{marker gene } 2 \\ \text{marker gene } 3 \\ \\ \text{marker gene } n \end{matrix}$$

Local point cloud

$$P = \{p_1, \ldots, p_m\} \subset \mathbb{R}^2$$

2. Rips persistence for marker gene correlations

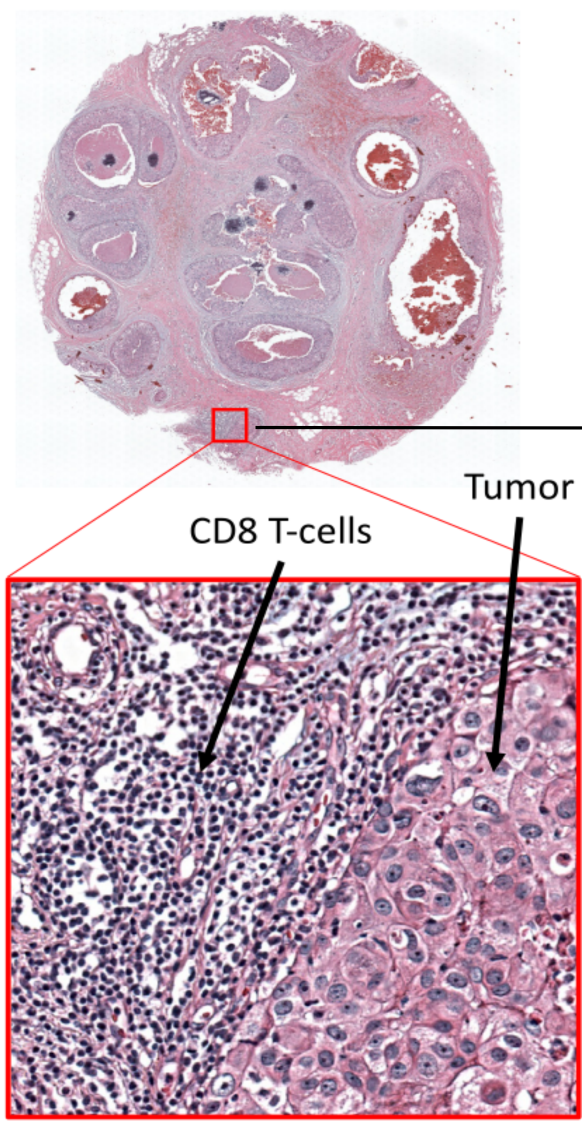
Q: What are the relations and correlations between *spatial features* and *marker genes*? Can one influence the other?



2. Rips persistence for marker gene correlations

Q: What are the relations and correlations between *spatial features* and *marker genes*? Can one influence the other?

A: Use *Pearson correlation* between marker gene expression and **Rips persistence** of local point clouds.



$$\begin{bmatrix} 0.2 \\ 0.4 \\ 0.5 \\ \vdots \\ 0.1 \end{bmatrix}$$

marker gene 1
marker gene 2
marker gene 3
marker gene n

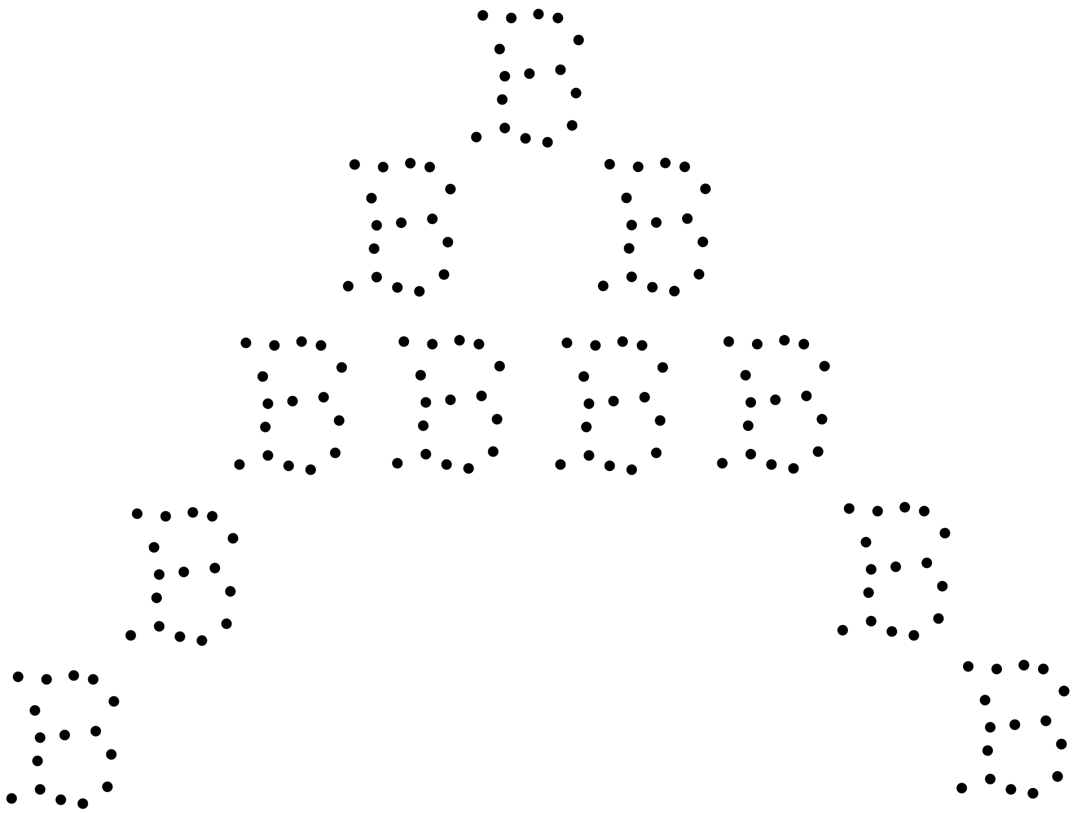
Local point cloud

$$P = \{p_1, \dots, p_m\} \subset \mathbb{R}^2$$

correlation?

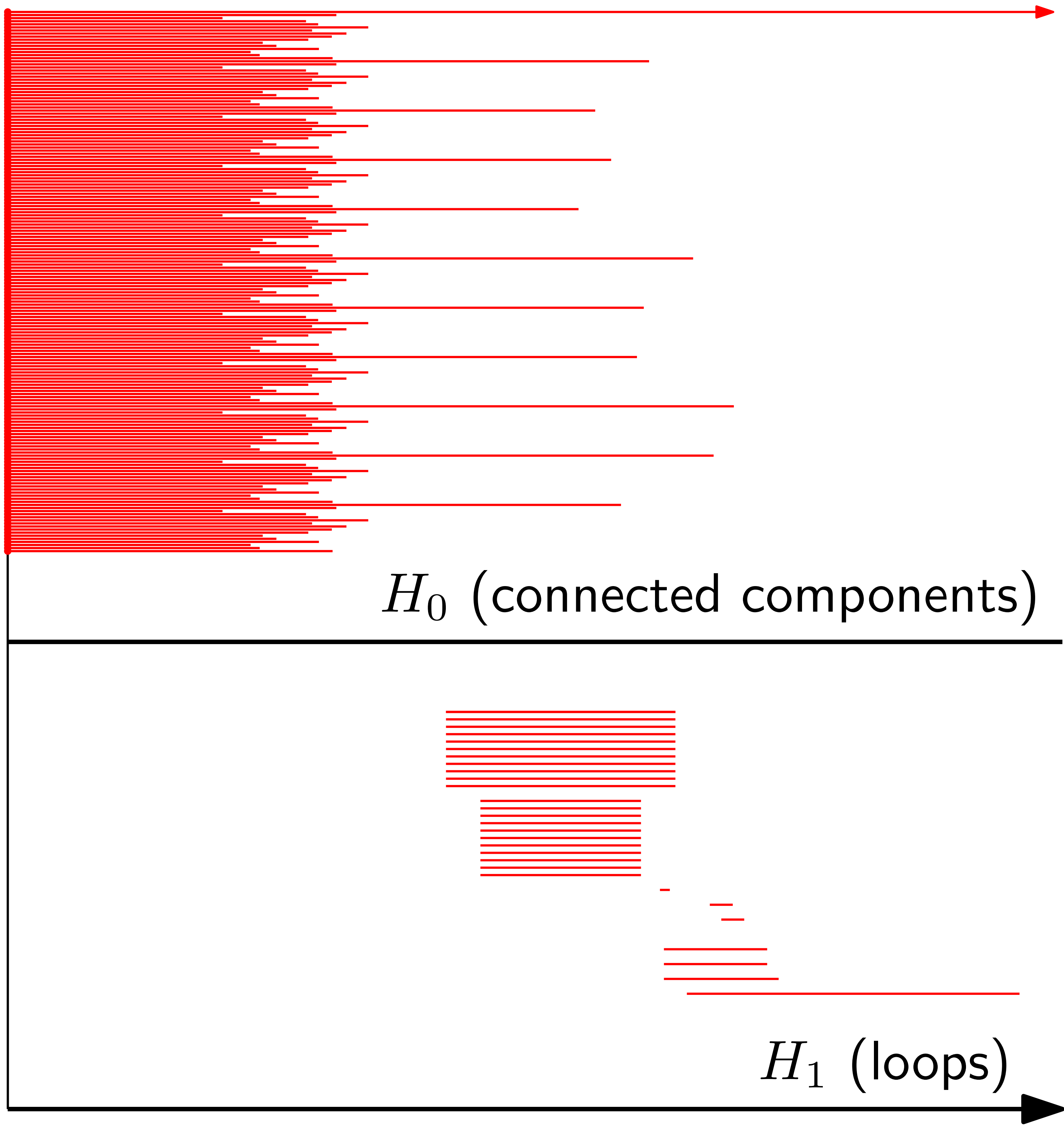
0- and 1-dimensional PH of Čech complexes

$X = \mathbb{R}^2$



sublevel sets of f = distance to a (pre-defined) point cloud P

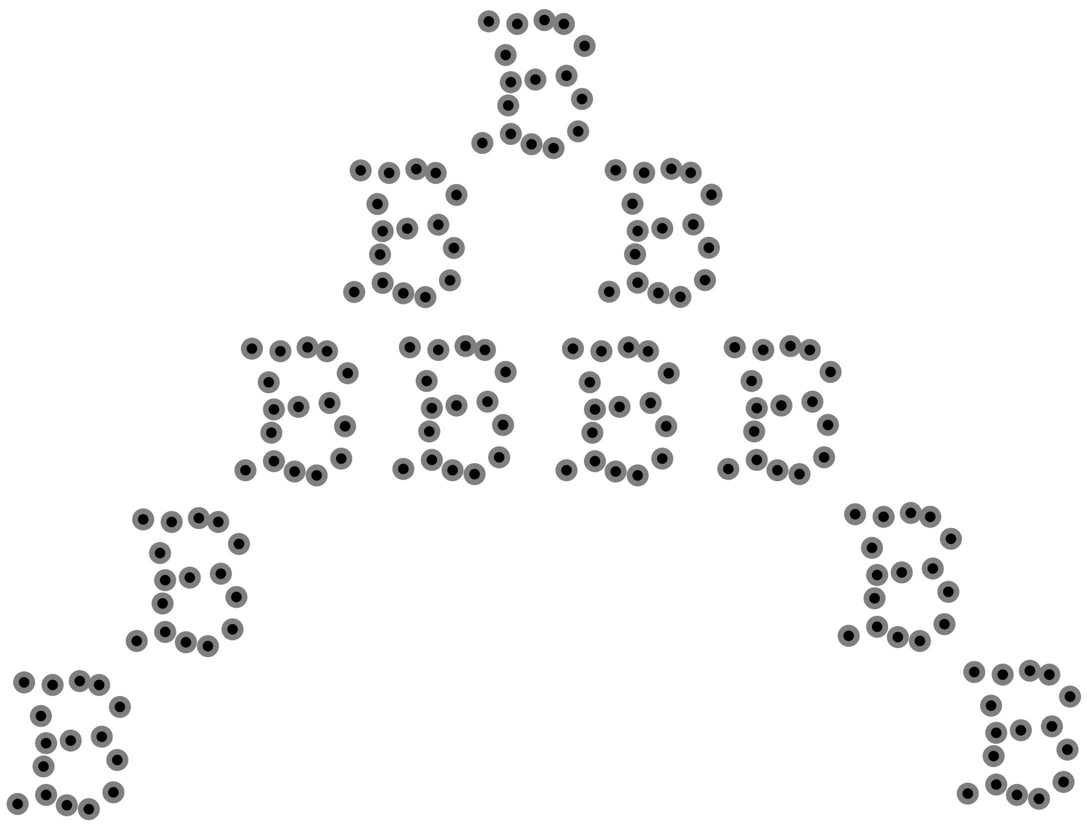
$f(x) := \min_{p \in P} \|x - p\|$



Persistence barcode

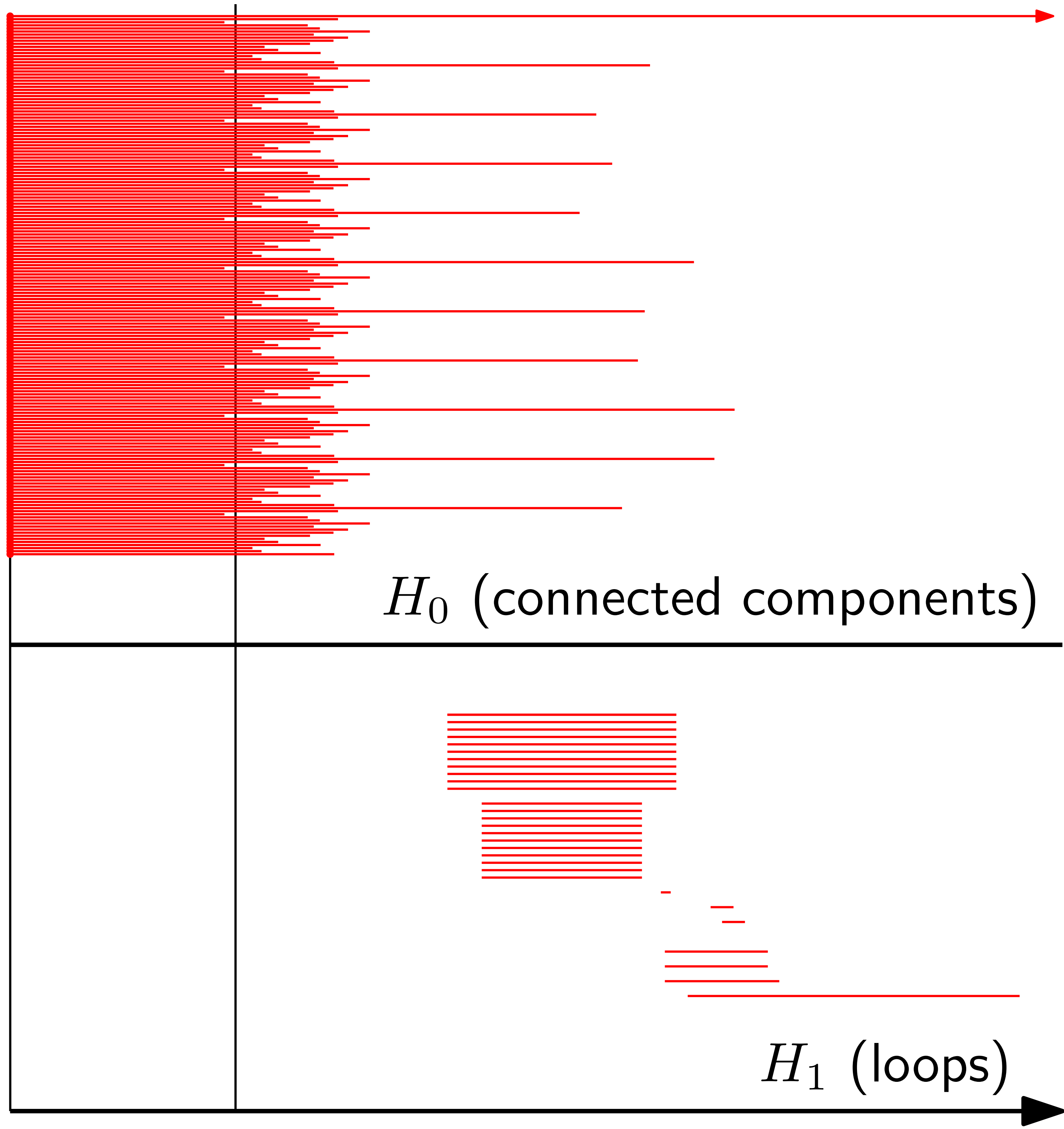
0- and 1-dimensional PH of Čech complexes

$X = \mathbb{R}^2$



sublevel sets of f = distance to a (pre-defined) point cloud P

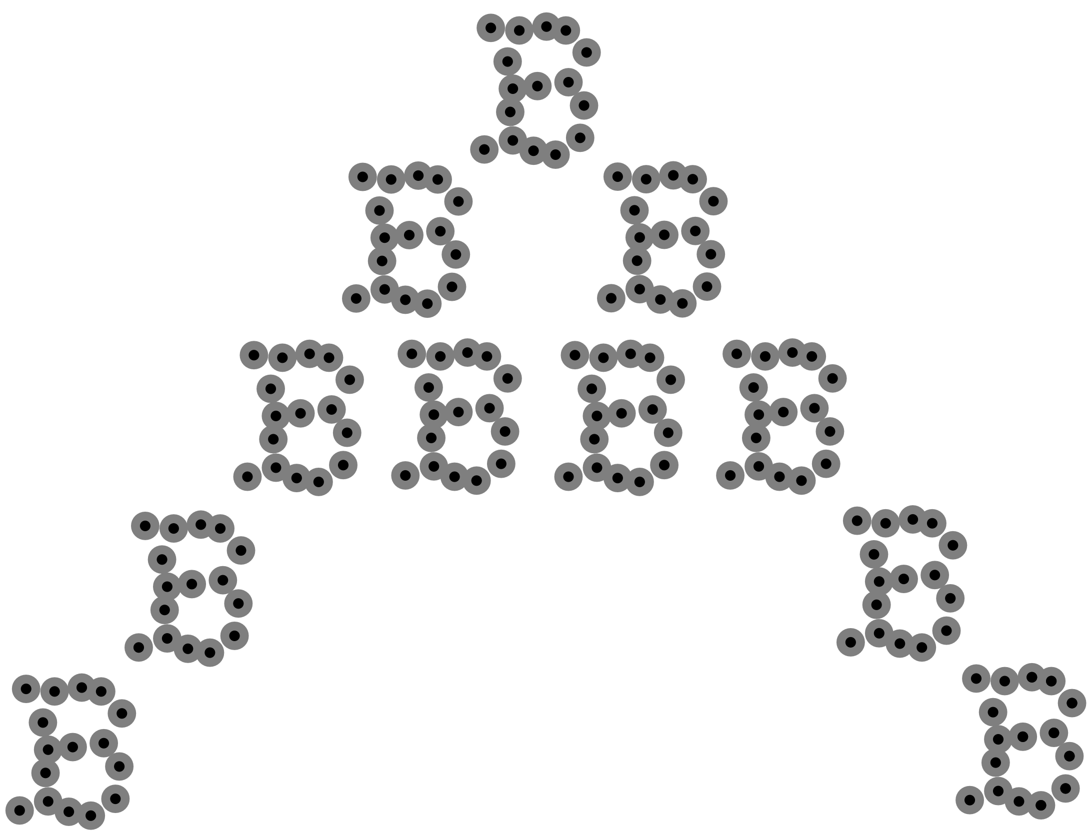
$f(x) := \min_{p \in P} \|x - p\|$



Persistence barcode

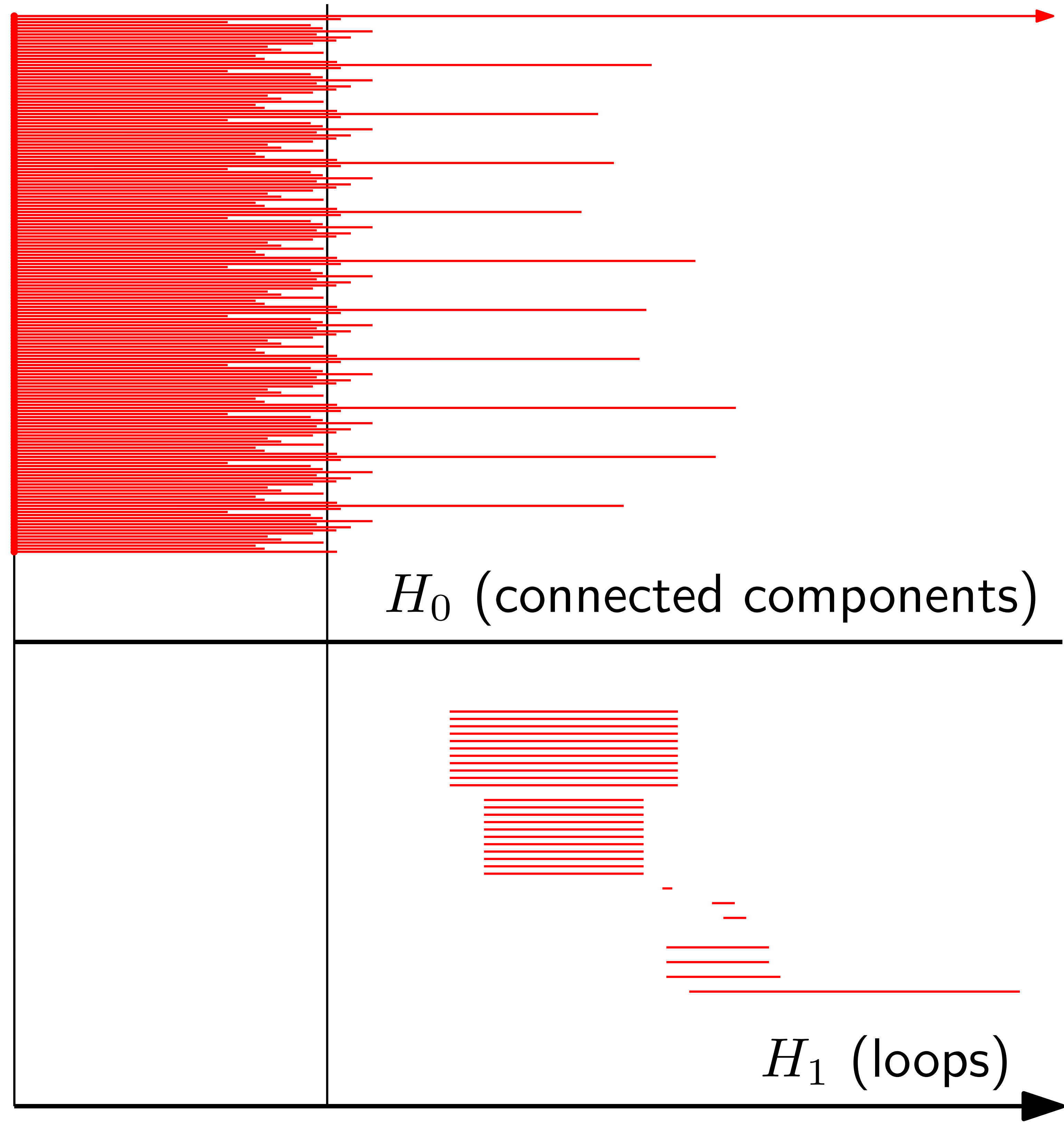
0- and 1-dimensional PH of Čech complexes

$X = \mathbb{R}^2$



sublevel sets of f = distance to a (pre-defined) point cloud P

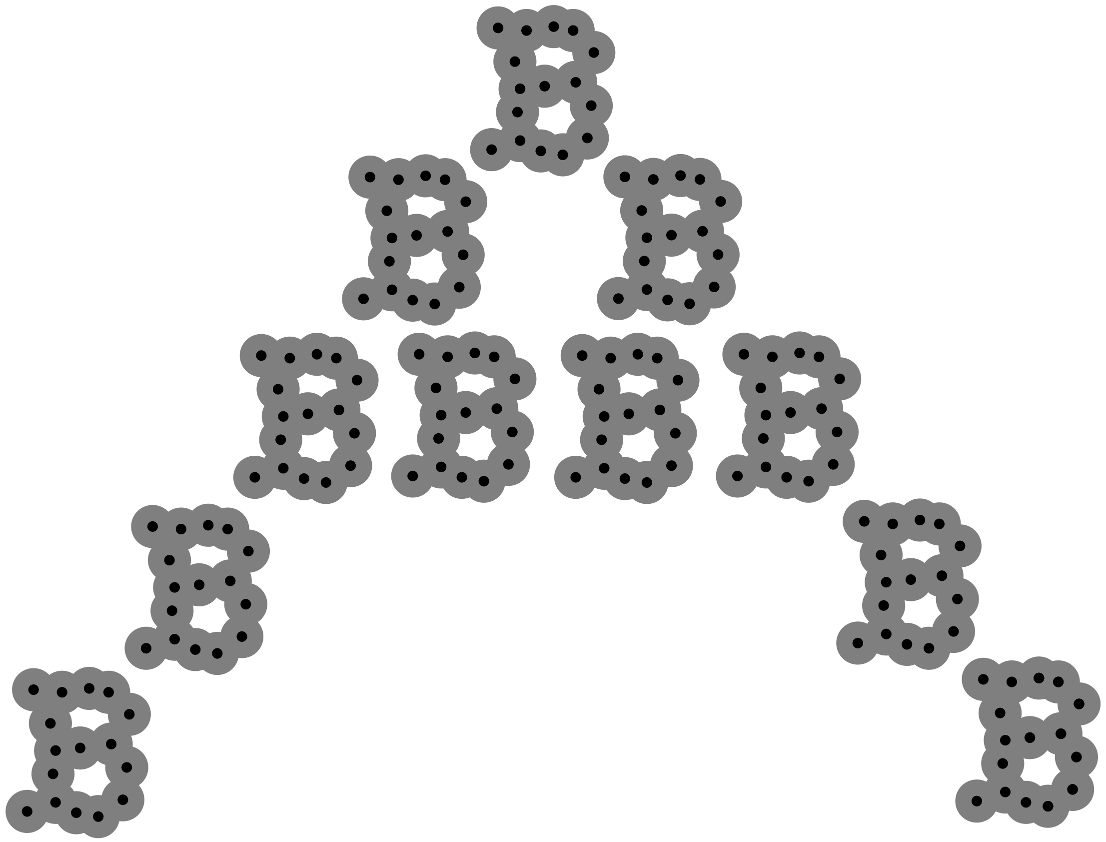
$$f(x) := \min_{p \in P} \|x - p\|$$



Persistence barcode

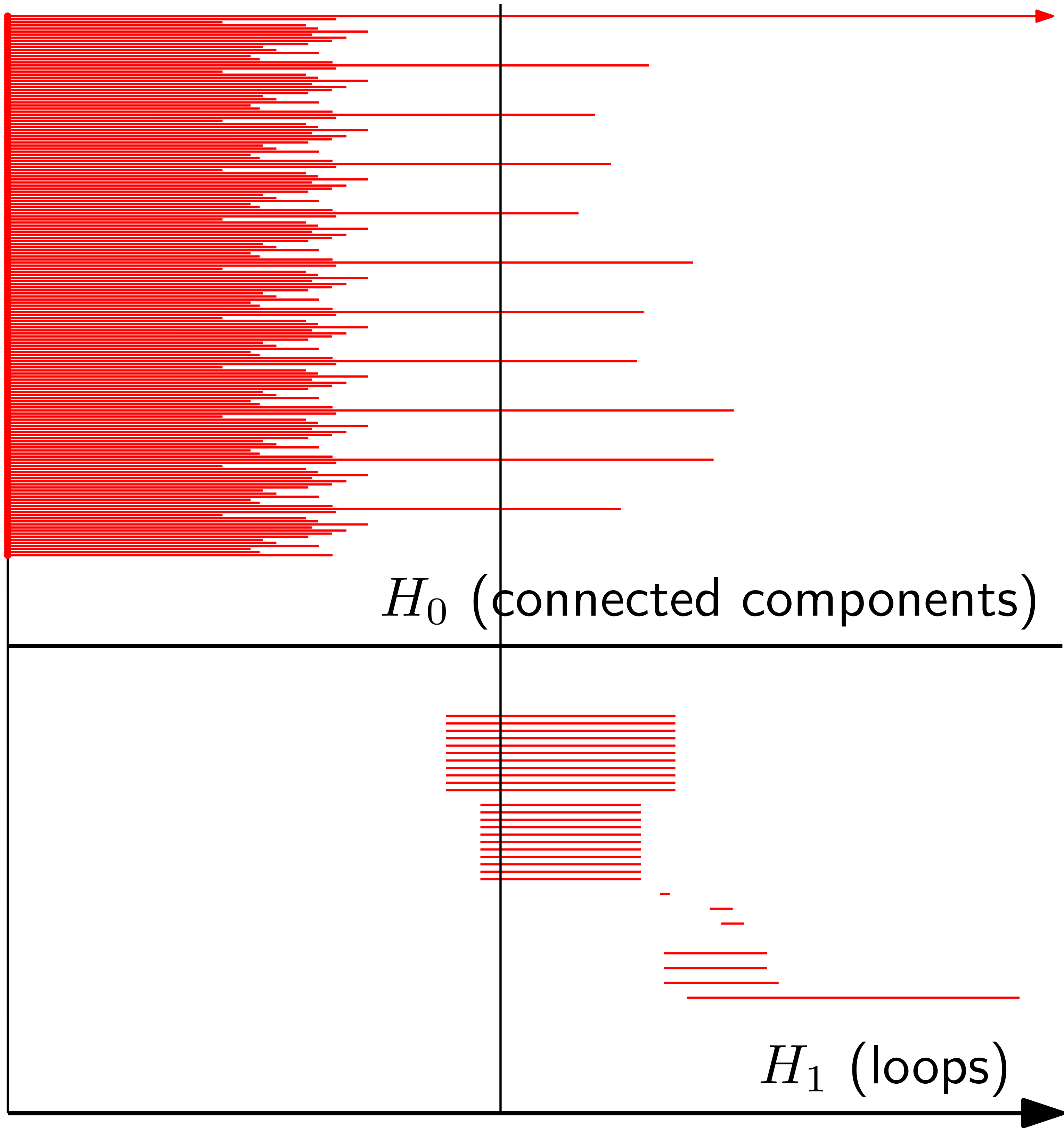
0- and 1-dimensional PH of Čech complexes

$X = \mathbb{R}^2$



sublevel sets of $f =$ distance to a (pre-defined) point cloud P

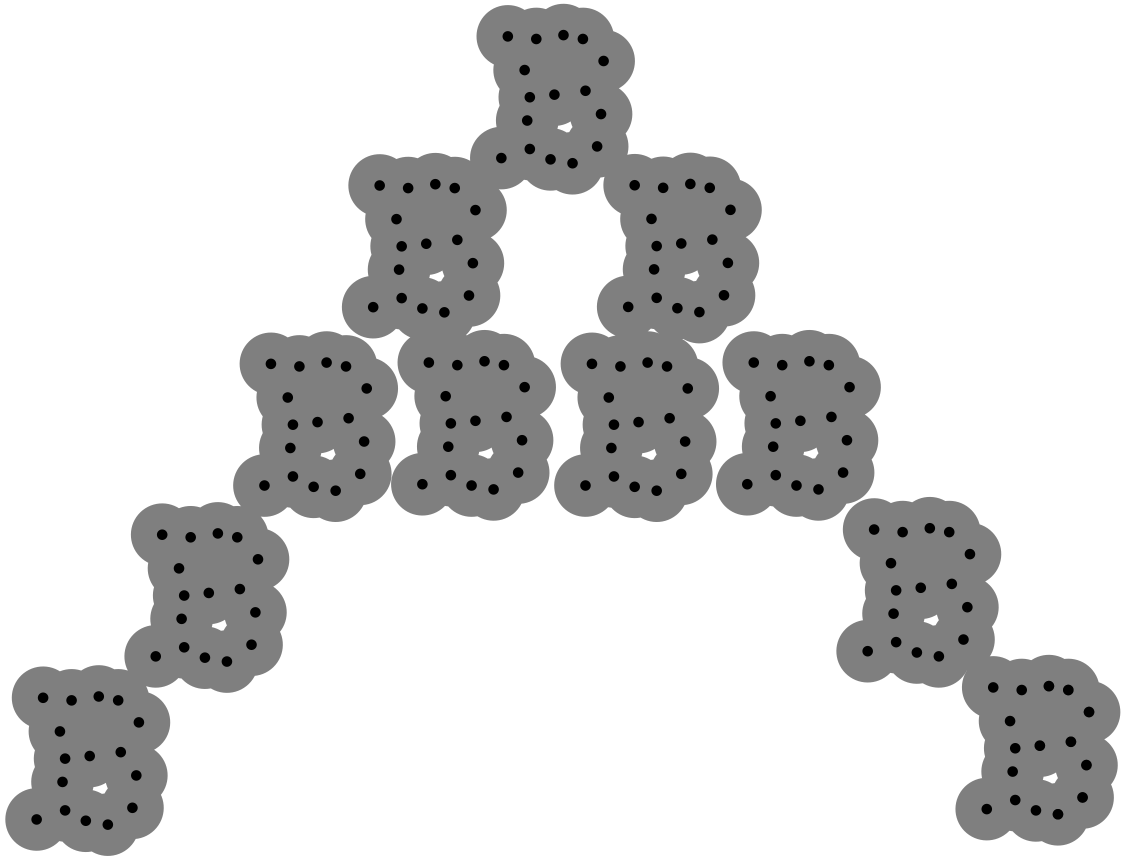
$$f(x) := \min_{p \in P} \|x - p\|$$



Persistence barcode

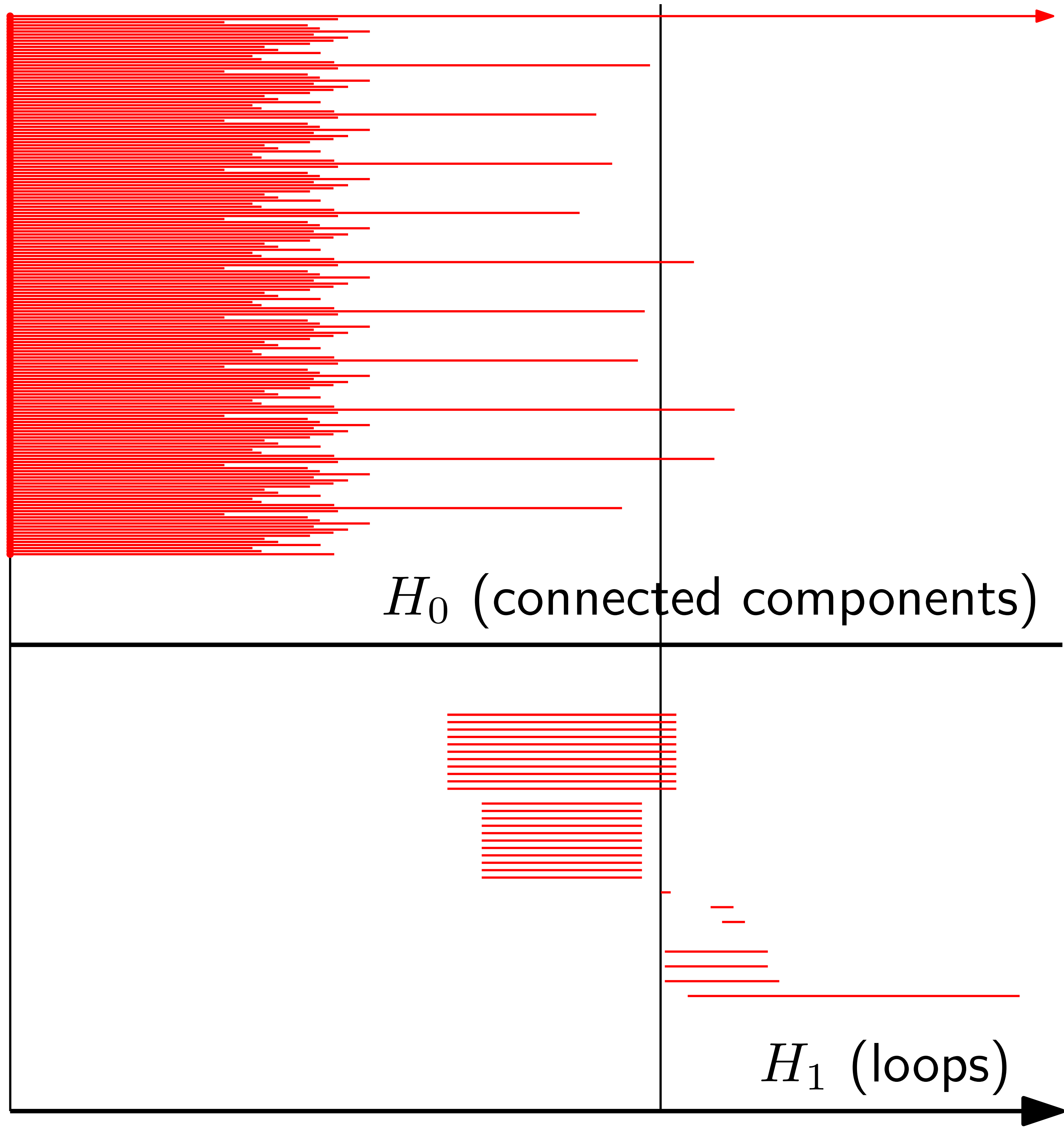
0- and 1-dimensional PH of Čech complexes

$X = \mathbb{R}^2$



sublevel sets of f = distance to a (pre-defined) point cloud P

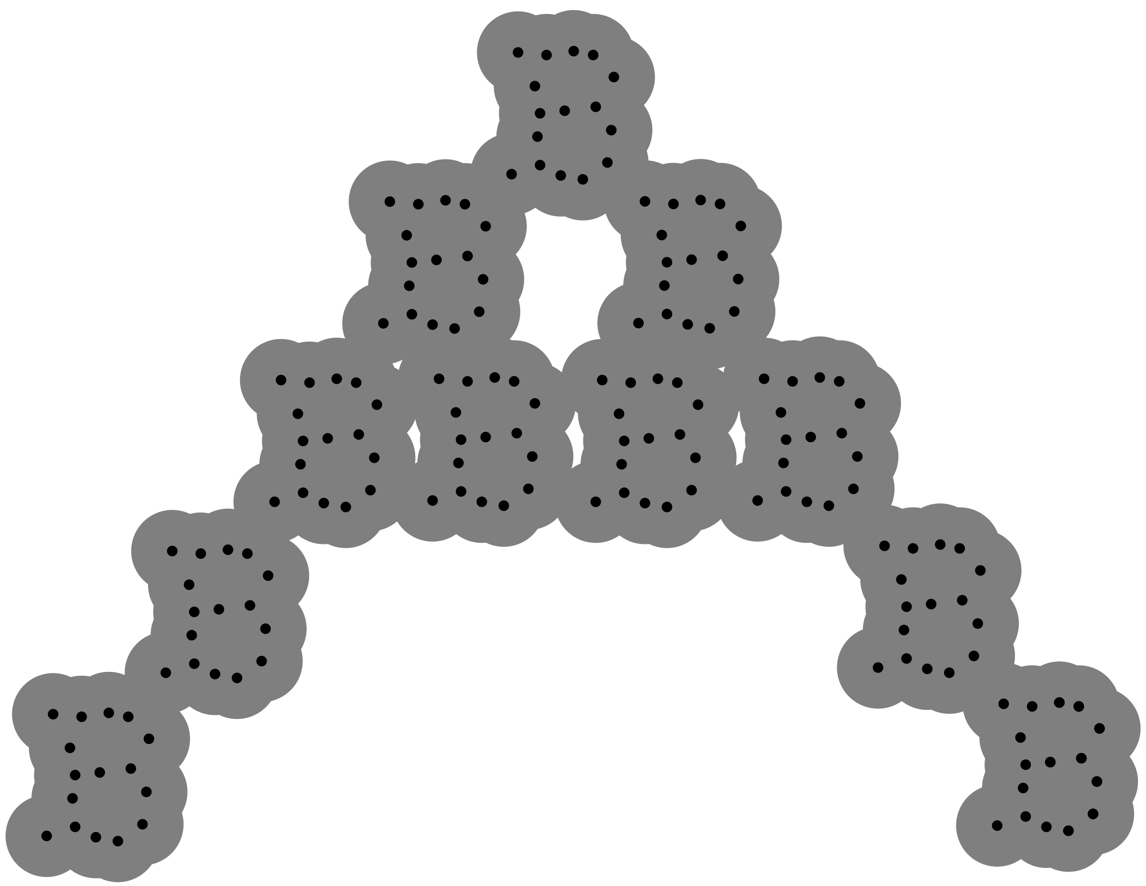
$$f(x) := \min_{p \in P} \|x - p\|$$



Persistence barcode

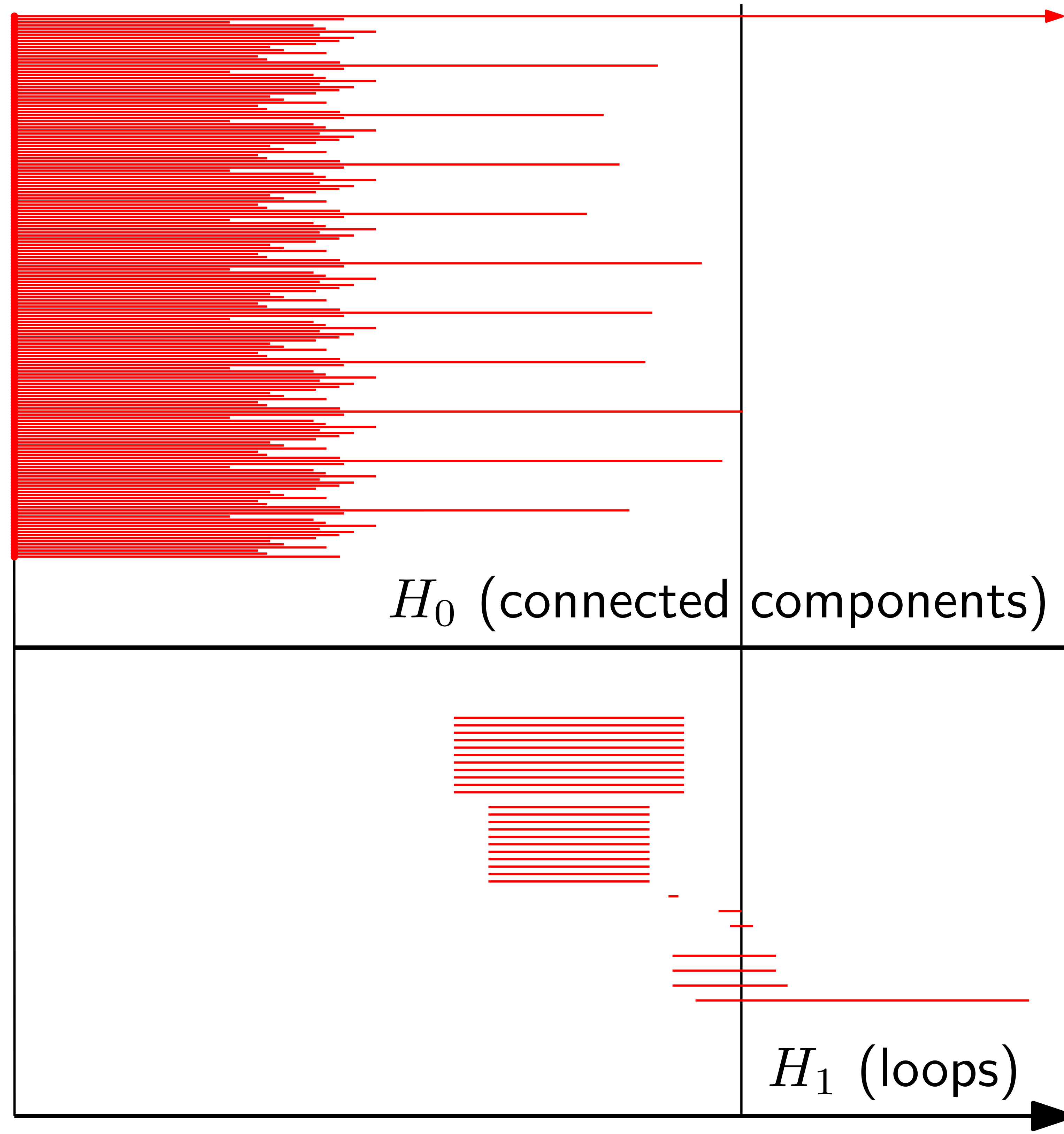
0- and 1-dimensional PH of Čech complexes

$$X = \mathbb{R}^2$$



sublevel sets of f = distance to a (pre-defined) point cloud P

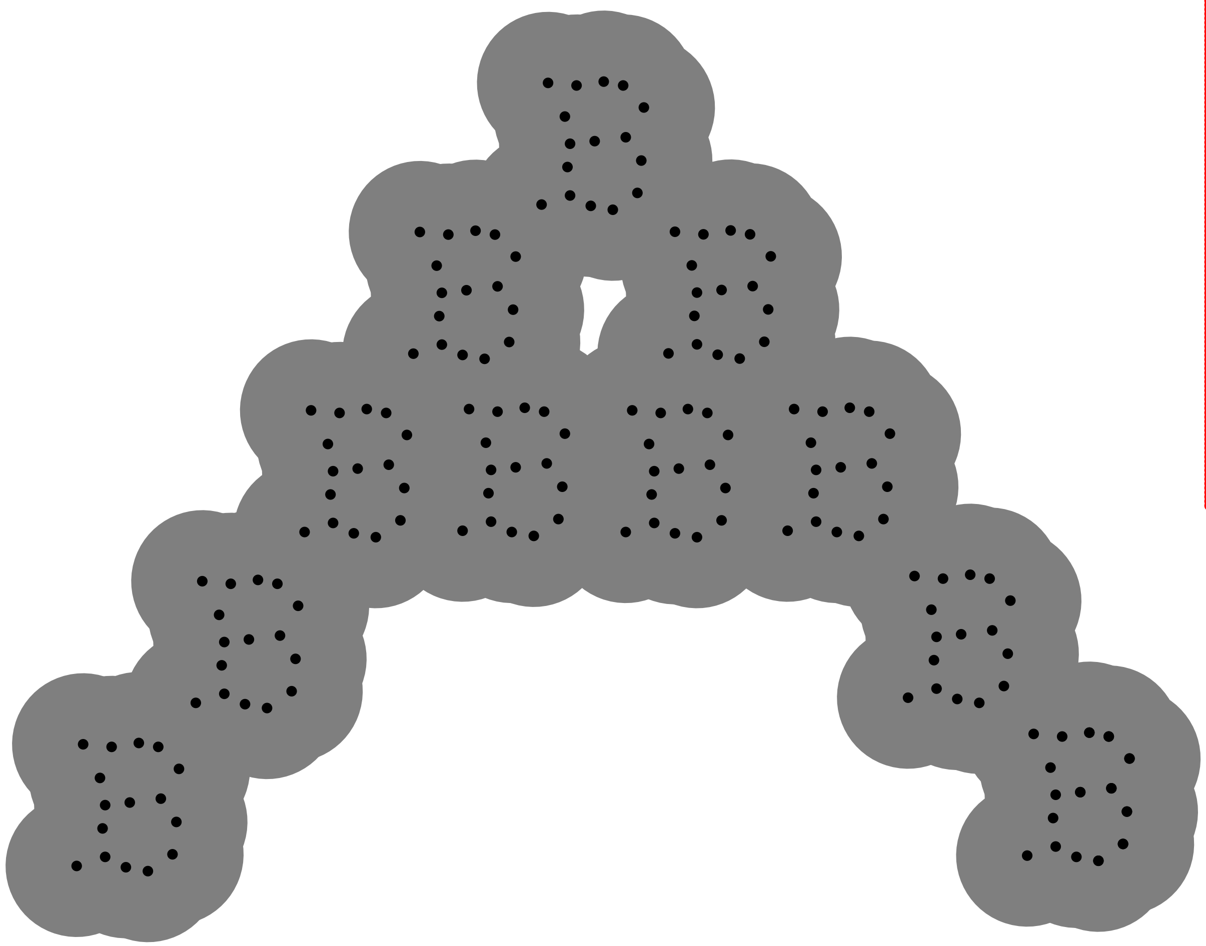
$$f(x) := \min_{p \in P} \|x - p\|$$



Persistence barcode

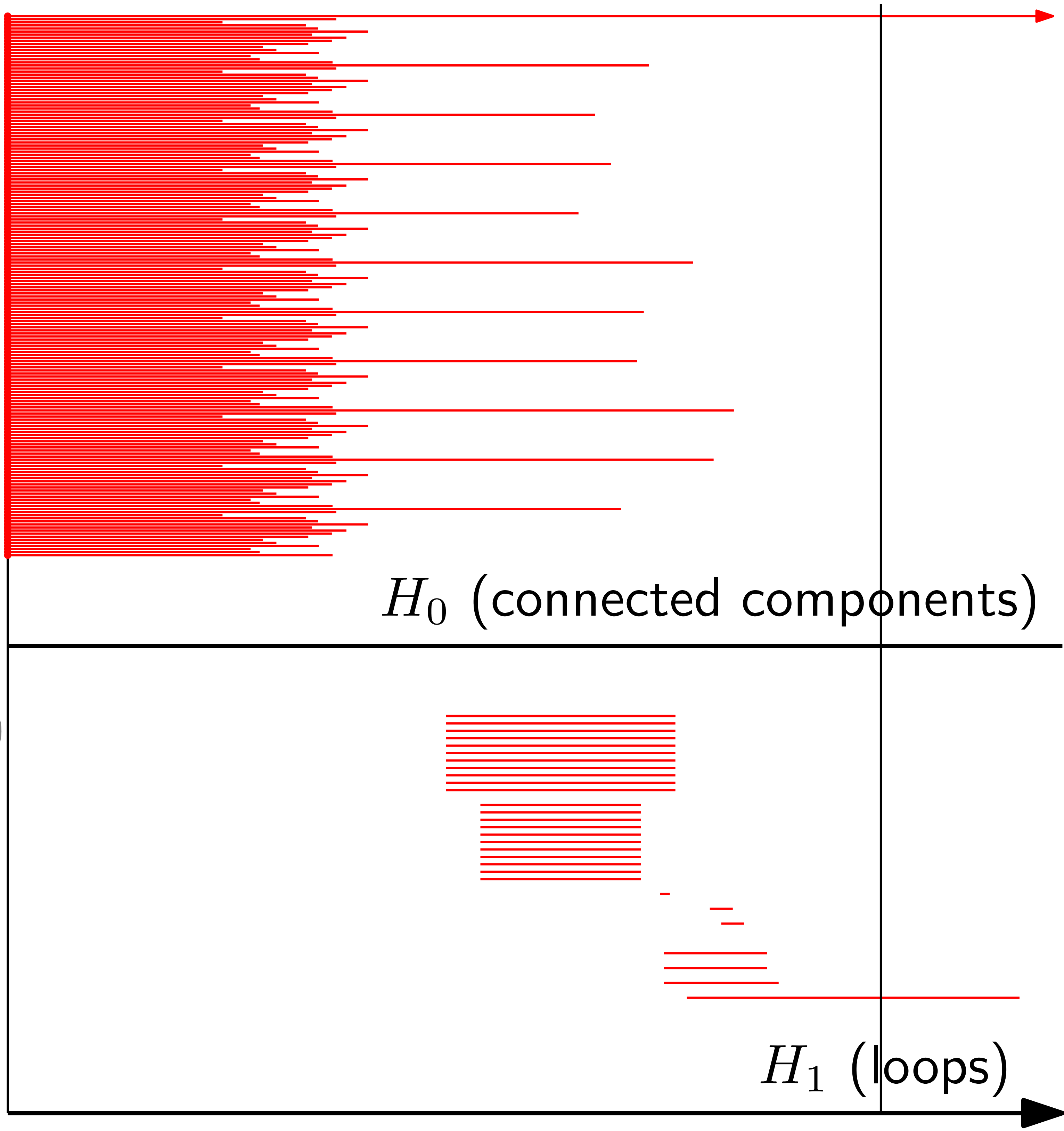
0- and 1-dimensional PH of Čech complexes

$X = \mathbb{R}^2$



sublevel sets of f = distance to a (pre-defined) point cloud P

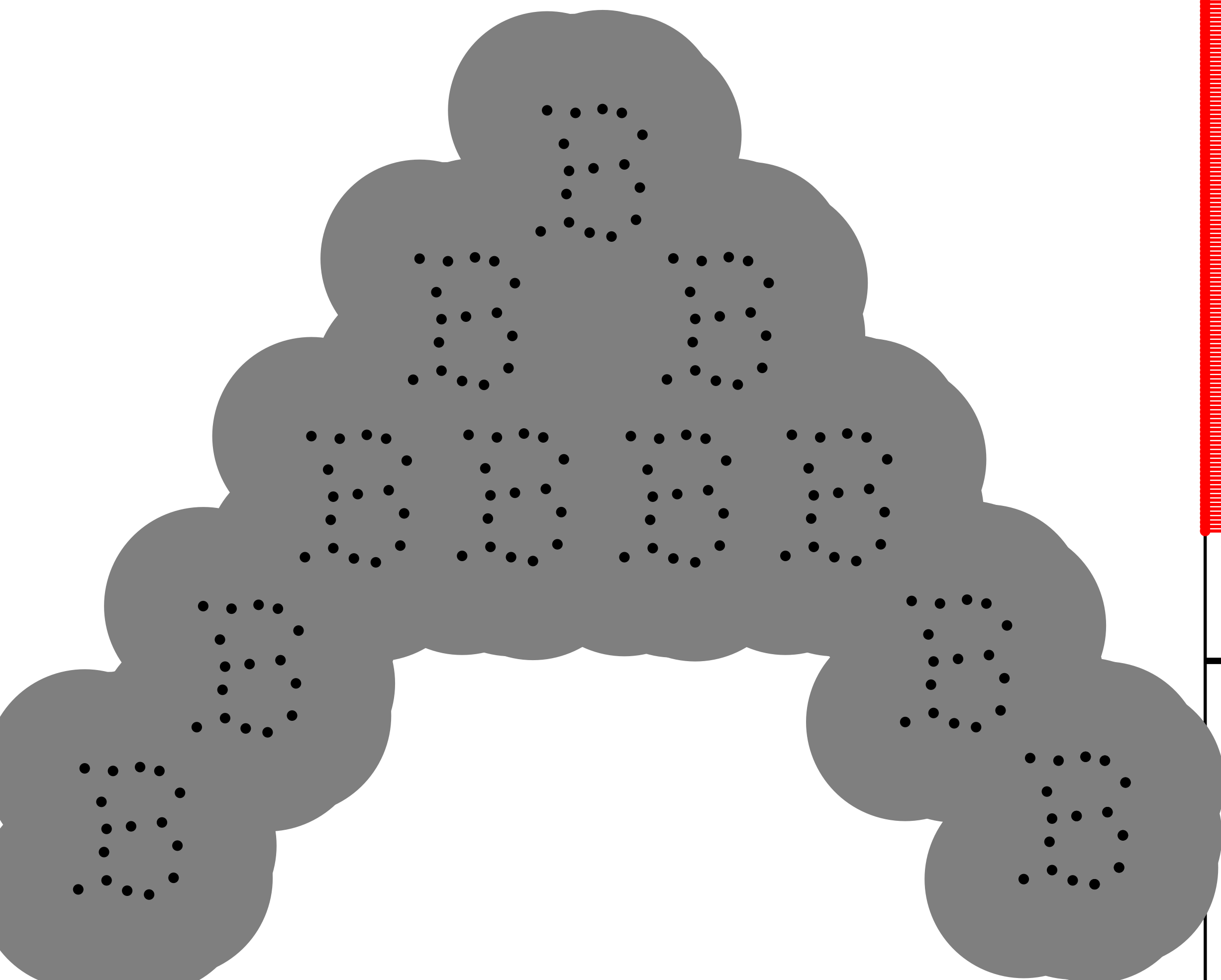
$$f(x) := \min_{p \in P} \|x - p\|$$



Persistence barcode

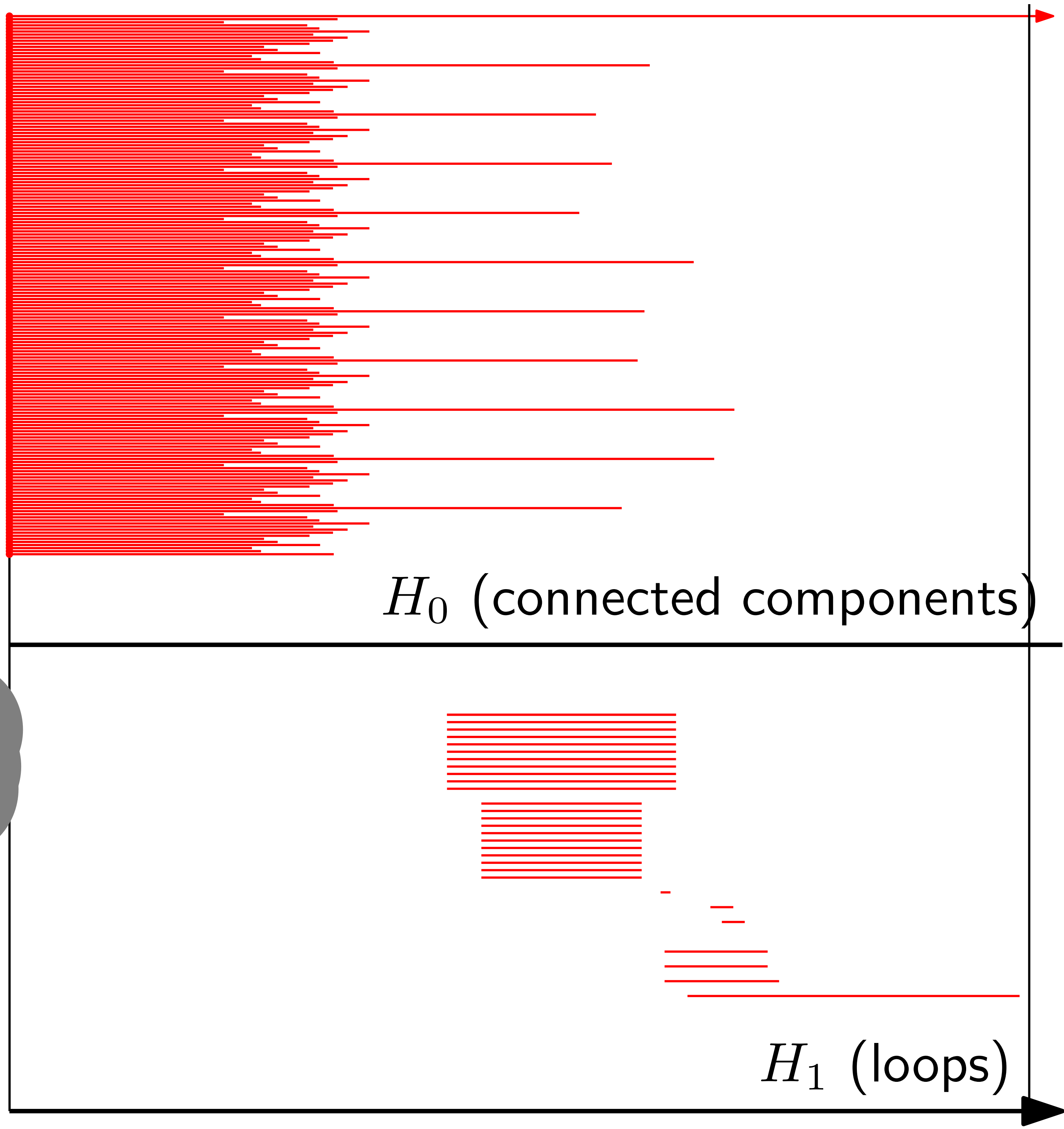
0- and 1-dimensional PH of Čech complexes

$X = \mathbb{R}^2$



sublevel sets of f = distance to a (pre-defined) point cloud P

$$f(x) := \min_{p \in P} \|x - p\|$$



Persistence barcode

Čech and (Vietoris)-Rips complexes

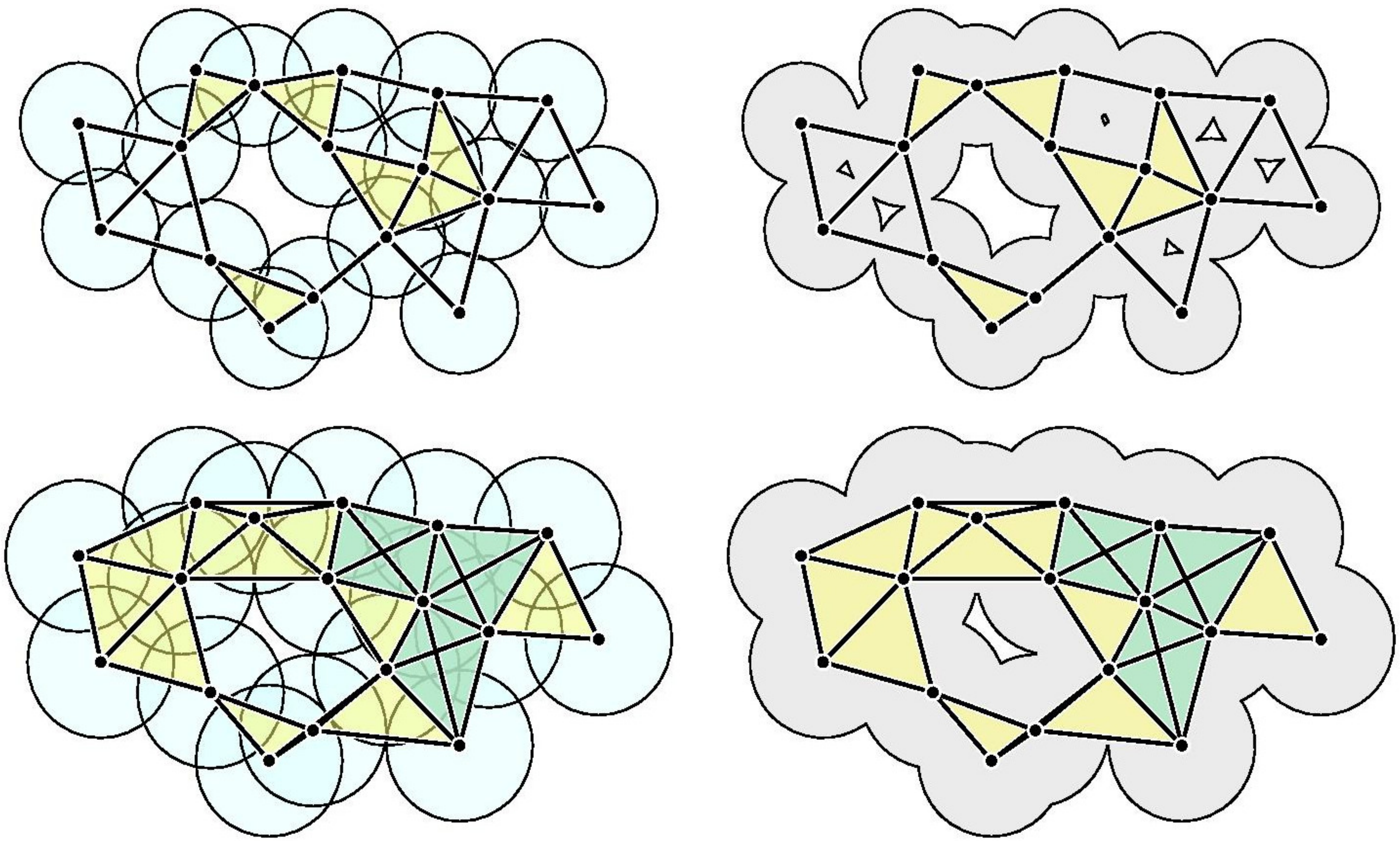
Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P, r)$ s.t. $\mathrm{vert}(C(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P, r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$

Čech and (Vietoris)-Rips complexes

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P, r)$ s.t. $\mathrm{vert}(C(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P, r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$



Čech and (Vietoris)-Rips complexes

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P, r)$ s.t. $\mathrm{vert}(C(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P, r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$

Pbm: Čech complexes can be quite hard to compute.

Čech and (Vietoris)-Rips complexes

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P, r)$ s.t. $\mathrm{vert}(C(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P, r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$

Pbm: Čech complexes can be quite hard to compute.

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Rips complex of radius $r > 0$ is the abstract simplicial complex $R(P, r)$ s.t. $\mathrm{vert}(R(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in R(P, r) \quad \text{iif} \quad \|P_{i_j} - P_{i_{j'}}\| \leq 2r, \forall 1 \leq j, j' \leq k.$$

Čech and (Vietoris)-Rips complexes

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P, r)$ s.t. $\mathrm{vert}(C(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P, r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$

Pbm: Čech complexes can be quite hard to compute.

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Rips complex of radius $r > 0$ is the abstract simplicial complex $R(P, r)$ s.t. $\mathrm{vert}(R(P, r)) = P$ and

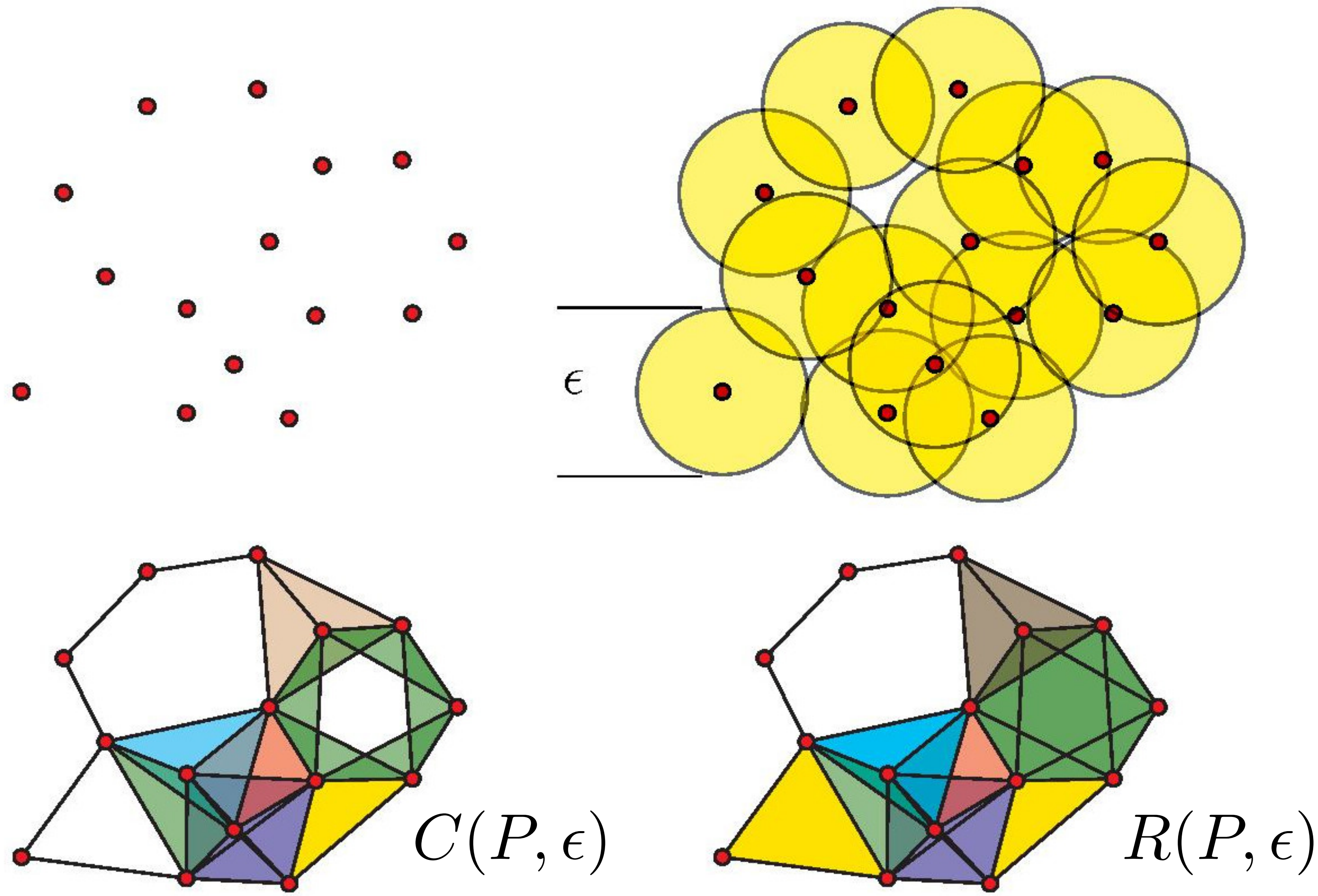
$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in R(P, r) \quad \text{iif} \quad \|P_{i_j} - P_{i_{j'}}\| \leq 2r, \forall 1 \leq j, j' \leq k.$$

Remark: The 1-skeleton $\mathrm{Skel}_1(R(P, r))$ of a Rips complex of radius r is also called the r-*neighborhood graph* of P.

Čech and (Vietoris)-Rips complexes

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P, r)$ s.t. $\mathrm{vert}(C(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P, r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$



Čech and (Vietoris)-Rips complexes

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P,r)$ s.t. $\mathrm{vert}(C(P,r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P,r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$

Pbm: Čech complexes can be quite hard to compute.

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Rips complex of radius $r > 0$ is the abstract simplicial complex $R(P,r)$ s.t. $\mathrm{vert}(R(P,r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in R(P,r) \quad \text{iif} \quad \|P_{i_j} - P_{i_{j'}}\| \leq 2r, \forall 1 \leq j, j' \leq k.$$

Good news is that Rips and Čech complexes are related:

Čech and (Vietoris)-Rips complexes

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P, r)$ s.t. $\mathrm{vert}(C(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P, r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$

Pbm: Čech complexes can be quite hard to compute.

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Rips complex of radius $r > 0$ is the abstract simplicial complex $R(P, r)$ s.t. $\mathrm{vert}(R(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in R(P, r) \quad \text{iif} \quad \|P_{i_j} - P_{i_{j'}}\| \leq 2r, \forall 1 \leq j, j' \leq k.$$

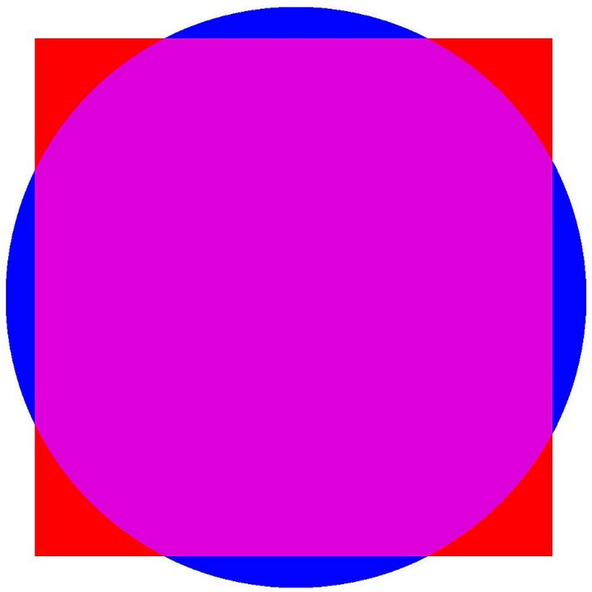
Good news is that Rips and Čech complexes are related:

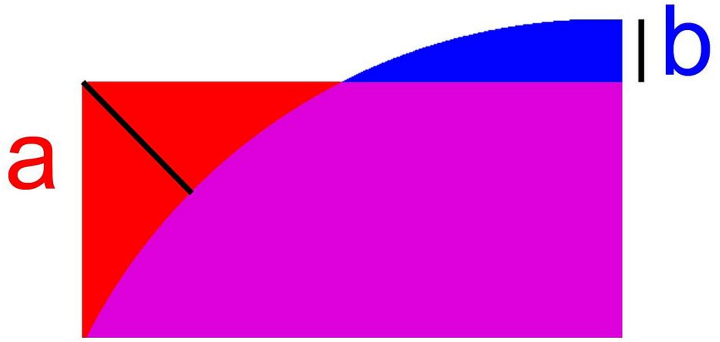
Prop: $R(P, r/2) \subseteq C(P, r) \subseteq R(P, r)$.

Stability properties for point clouds

Def: The Hausdorff distance between two subspaces X, Y of a common metric space (Z, d) is:

$$\begin{aligned} d_H(X,Y) &= \max\{\sup_{y\in Y} d(y,X), \sup_{x\in X} d(x,Y)\} \\ &= \max\{\sup_{y\in Y}\inf_{x\in X} d(y,x), \sup_{x\in X}\inf_{y\in Y} d(x,y)\} \end{aligned}$$





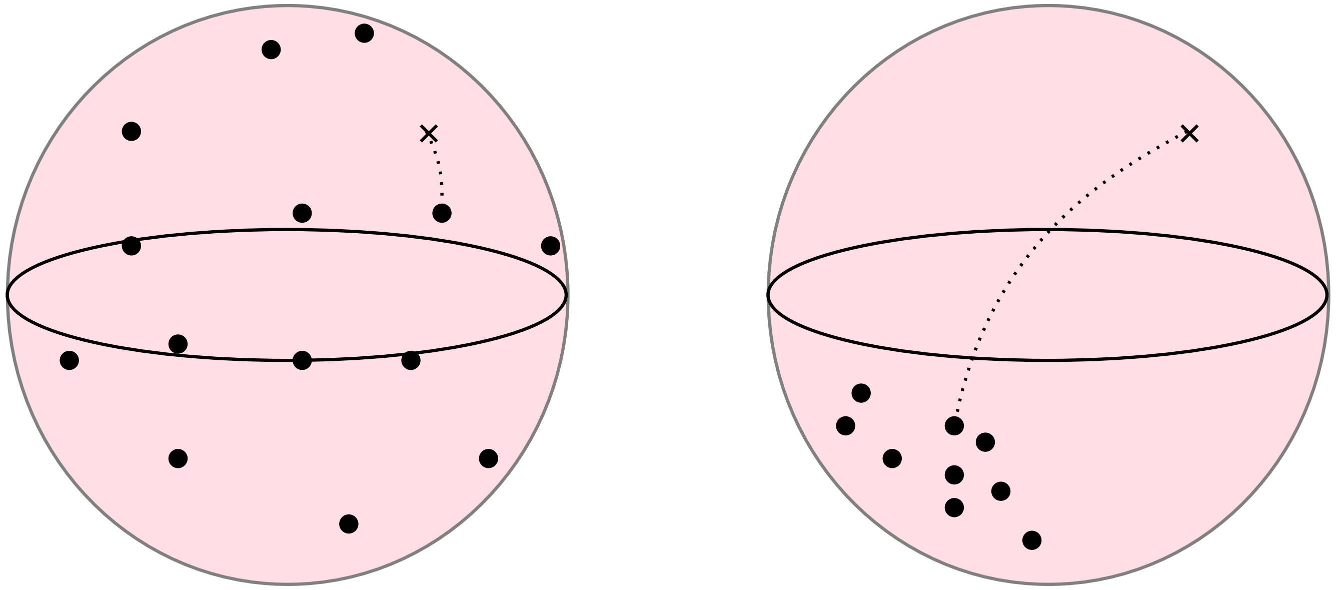
$$d_H(X,Y) = \max\{a, b\}$$

Stability properties for point clouds

Def: The Hausdorff distance between two subspaces X, Y of a common metric space (Z, d) is:

$$\begin{aligned} d_H(X,Y) &= \max\{\sup_{y\in Y} d(y,X), \sup_{x\in X} d(x,Y)\} \\ &= \max\{\sup_{y\in Y} \inf_{x\in X} d(y,x), \sup_{x\in X} \inf_{y\in Y} d(x,y)\} \end{aligned}$$

Ex: Given a sampling $\hat{X}_n \subseteq X$, $d_H(\hat{X}_n, X)$ is a measure of sampling quality.



Stability properties for point clouds

Def: The Hausdorff distance between two subspaces X, Y of a common metric space (Z, d) is:

$$d_H(X,Y) = \max\{\sup_{y\in Y} d(y,X), \sup_{x\in X} d(x,Y)\}$$
$$= \max\{\sup_{y\in Y}\inf_{x\in X} d(y,x), \sup_{x\in X}\inf_{y\in Y} d(x,y)\}$$

Def: The Gromov-Hausdorff distance between metric spaces $(X, d_X), (Y, d_Y)$ is the Hausdorff distance of the best common isometric embedding:

$$d_{GH}((X,d_X),(Y,d_Y)) = \inf_\gamma d_H(\gamma(X),\gamma(Y)),$$

where $d(\gamma(x),\gamma(x')) = d_X(x,x')$ and $d(\gamma(y),\gamma(y')) = d_X(y,y')$.

Stability properties for point clouds

Def: The Hausdorff distance between two subspaces X, Y of a common metric space (Z, d) is:

$$d_H(X,Y) = \max\{\sup_{y\in Y} d(y,X), \sup_{x\in X} d(x,Y)\}$$
$$= \max\{\sup_{y\in Y} \inf_{x\in X} d(y,x), \sup_{x\in X} \inf_{y\in Y} d(x,y)\}$$

Def: The Gromov-Hausdorff distance between metric spaces $(X, d_X), (Y, d_Y)$ is metric distortion of the best correspondence:

$$d_{GH}((X,d_X),(Y,d_Y)) = \inf_{\mathcal{C}} \sup_{(x,y),(x',y')\in\mathcal{C}} |d_X(x,x') - d_Y(y,y')|,$$

where $\mathcal{C} \subseteq X \times Y$ s.t. $\forall x$, $\exists y_x \in Y$ s.t. $(x, y_x) \in \mathcal{C}$ (and vice-versa).

Stability properties for point clouds

Def: The Hausdorff distance between two subspaces X, Y of a common metric space (Z, d) is:

$$d_H(X,Y) = \max\{\sup_{y\in Y} d(y,X), \sup_{x\in X} d(x,Y)\}$$
$$= \max\{\sup_{y\in Y}\inf_{x\in X} d(y,x), \sup_{x\in X}\inf_{y\in Y} d(x,y)\}$$

Def: The Gromov-Hausdorff distance between metric spaces $(X, d_X), (Y, d_Y)$ is metric distortion of the best correspondence:

$$d_{GH}((X,d_X),(Y,d_Y)) = \inf_{\mathcal{C}} \sup_{(x,y),(x',y')\in\mathcal{C}} |d_X(x,x') - d_Y(y,y')|,$$

where $\mathcal{C} \subseteq X \times Y$ s.t. $\forall x$, $\exists y_x \in Y$ s.t. $(x, y_x) \in \mathcal{C}$ (and vice-versa).

Thm: If X and Y are common subspaces of a common metric space (Z, d), then

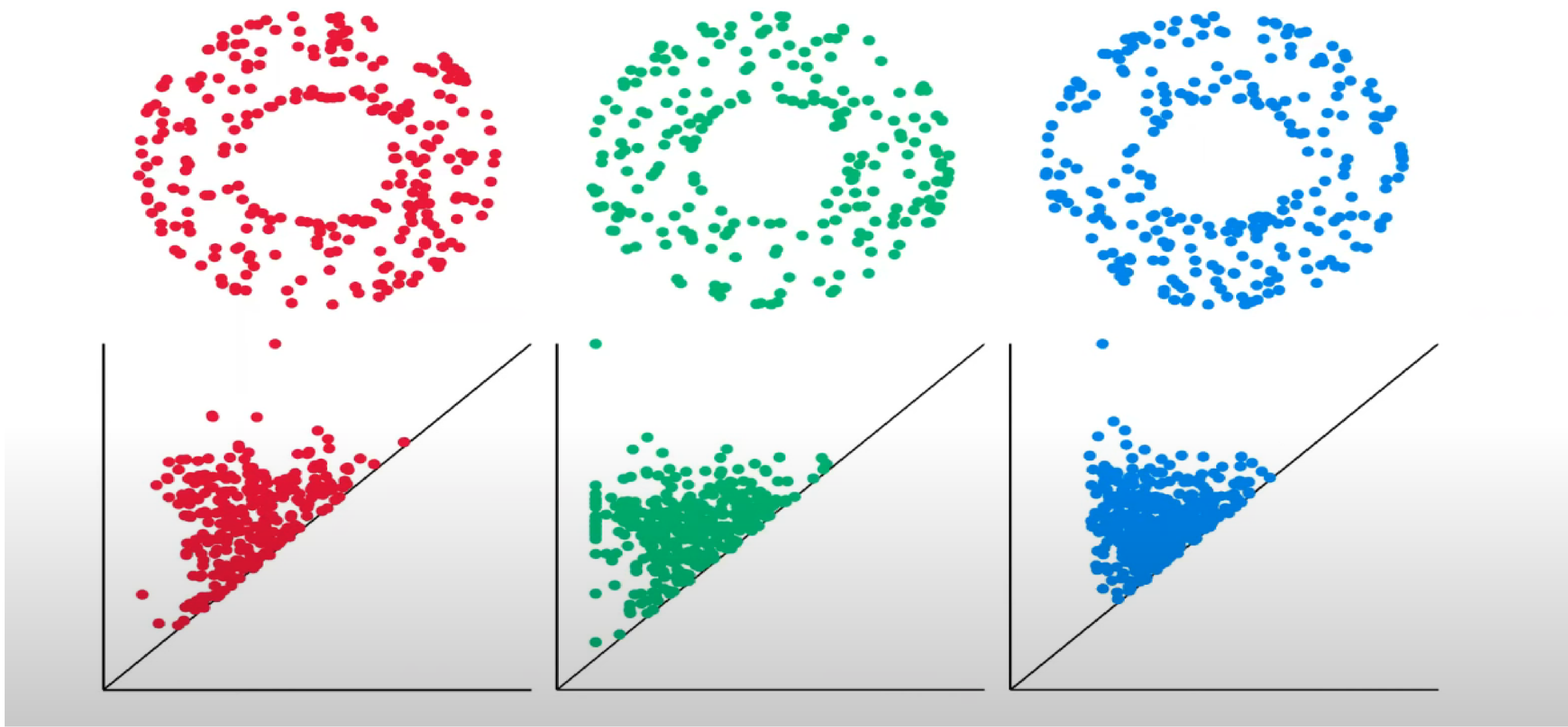
$$d_b(D_{\mathrm{Cech}}(X), D_{\mathrm{Cech}}(Y)) \leq d_H(X,Y).$$

Stability properties for point clouds

[*Persistence stability for geometric complexes*, Chazal, de Silva, Oudot, Geom. Dedicata, 2013].

Thm: If X and Y are pre-compact metric spaces, then

$$d_b(D_{\mathrm{Rips}}(X), D_{\mathrm{Rips}}(Y)) \leq d_{GH}(X, Y).$$



Rem: This result also holds for Čech and other families of filtrations (particular case of a more general theorem).

Representations of Persistence Diagrams

Representations of Persistence Diagrams

Q: Persistence diagrams are **not** Euclidean vectors? How can one compute Pearson correlation between marker gene expression and persistence diagrams?

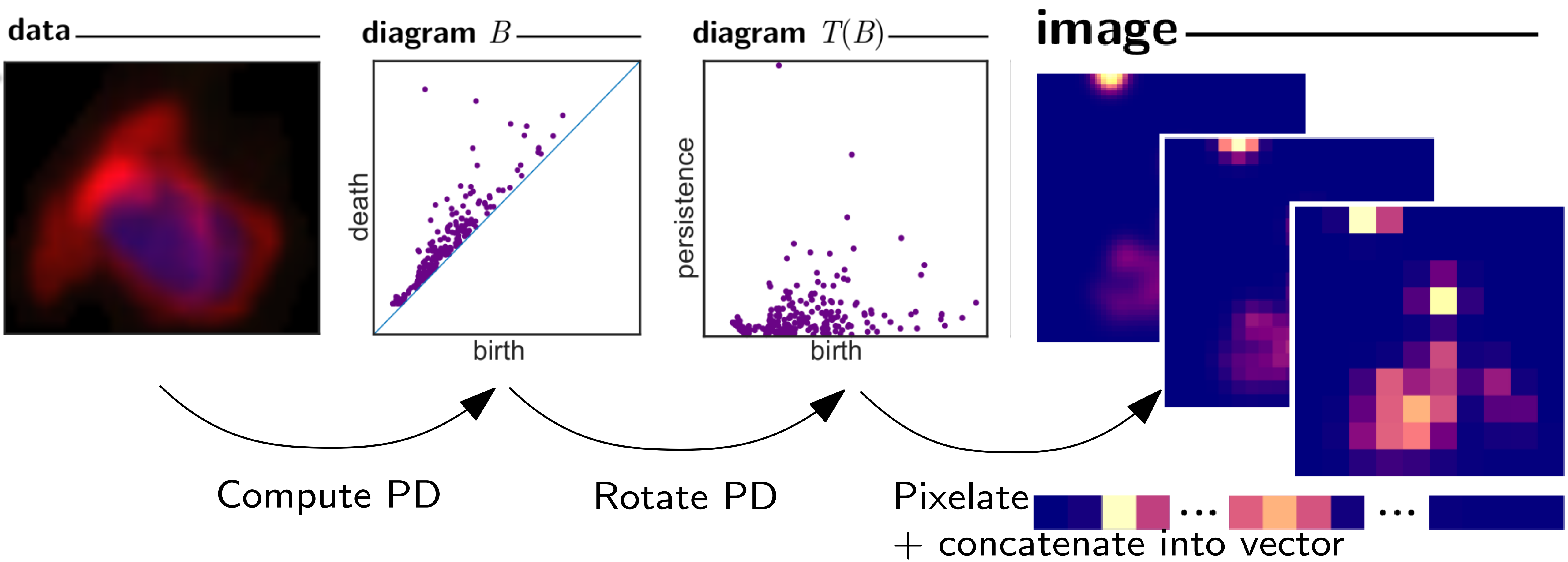
Representations of Persistence Diagrams

Q: Persistence diagrams are **not** Euclidean vectors? How can one compute Pearson correlation between marker gene expression and persistence diagrams?

A: Use **representations**, which are mappings $\Phi : \mathcal{D} \to \mathcal{H}$ from the space of persistence diagrams to Hilbert spaces.

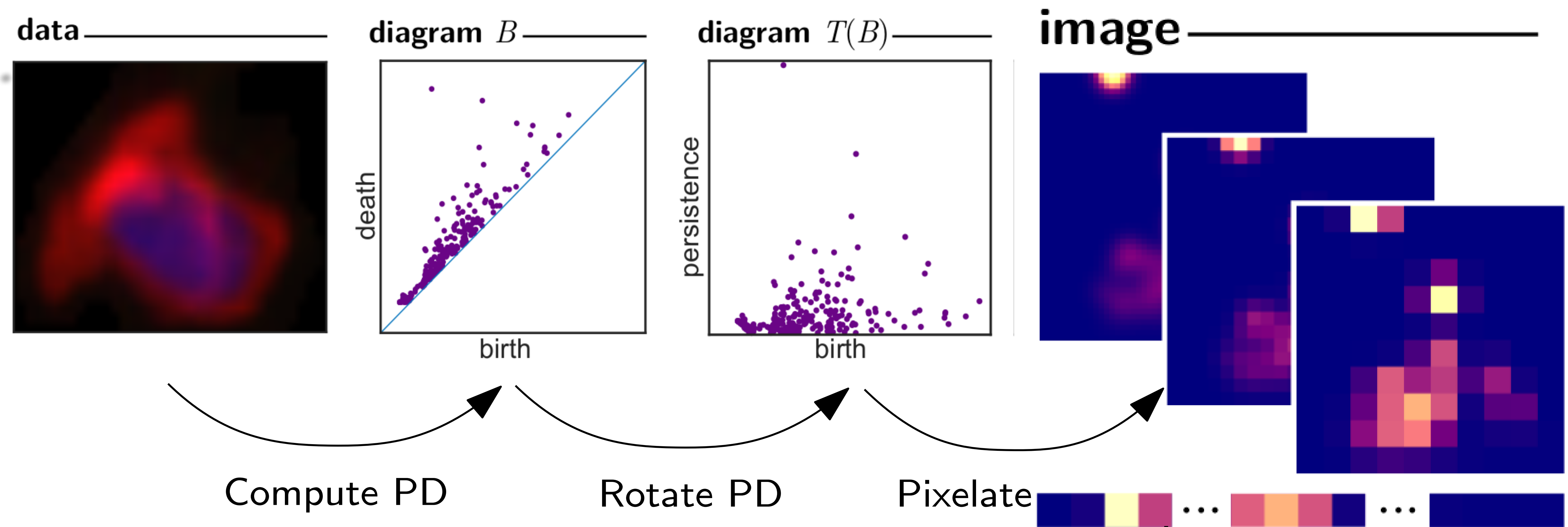
Persistence image

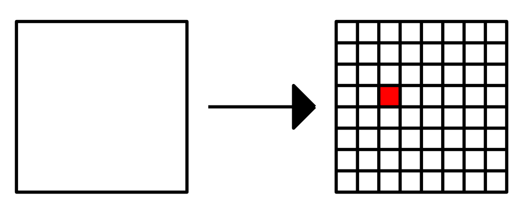
[*Persistence Images: A Stable Vector Representation of Persistent Homology*, Adams et al., JMLR, 2017]



Persistence image

[*Persistence Images: A Stable Vector Representation of Persistent Homology*, Adams et al., JMLR, 2017]



Discretize plane into a grid: 

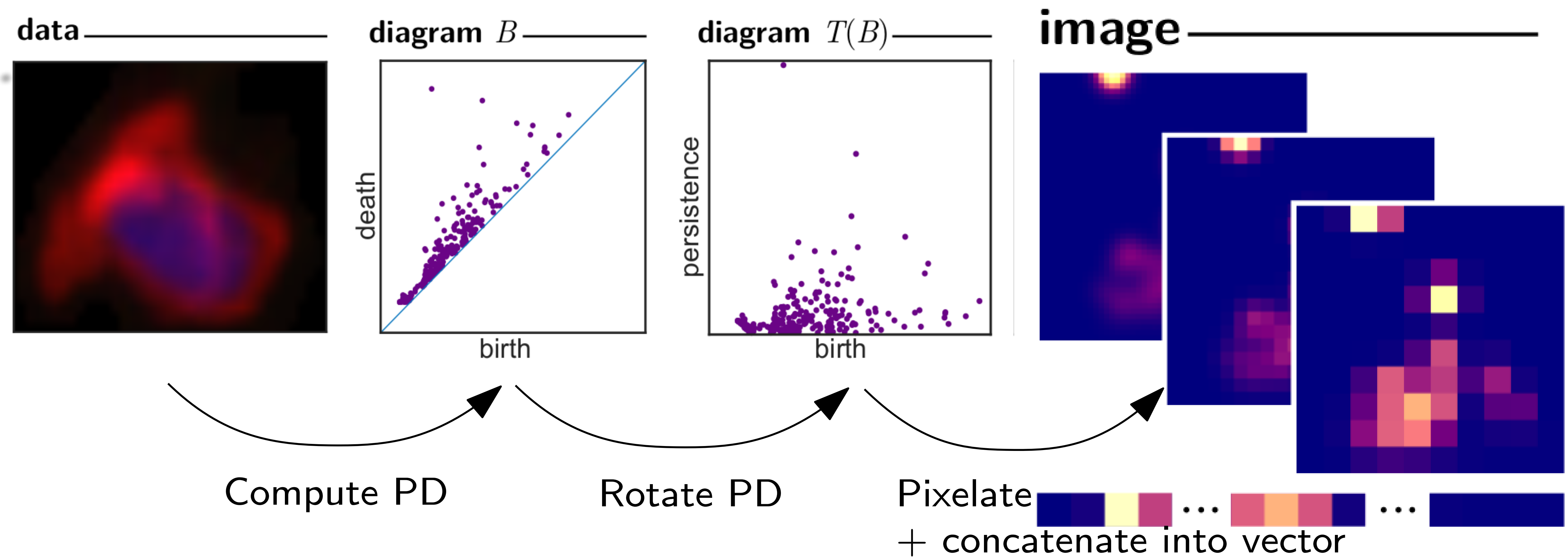
For each grid pixel P, compute $I(P) = \sum_{p \in D} \int\int_P w(p) \cdot \phi_p$.

Concatenate all $I(P)$ into a single vector $\mathrm{PI}(D)$.

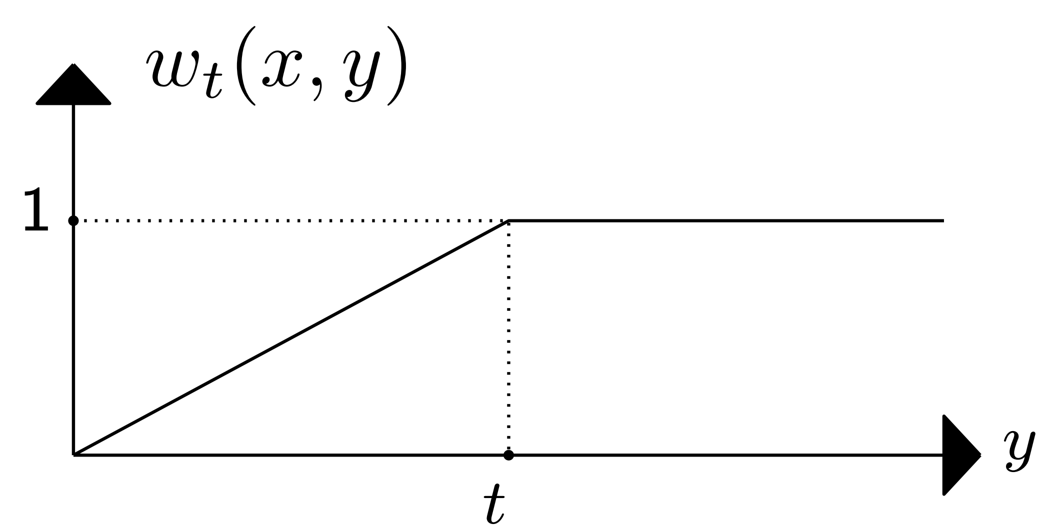
Ex: $\phi_p = \mathcal{N}(p, \sigma)$.

Persistence image

[*Persistence Images: A Stable Vector Representation of Persistent Homology*, Adams et al., JMLR, 2017]



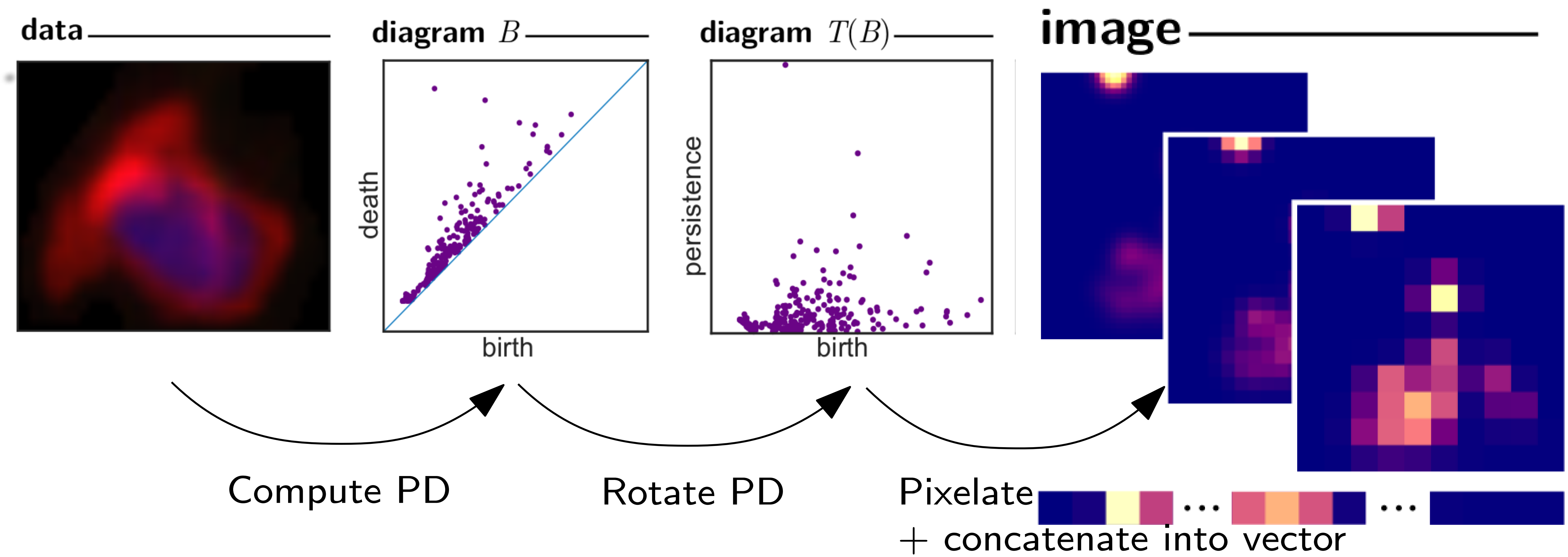
Weight functions that preserve stability must satisfy $w(p) \to 0$ when $d(p, \Delta) \to 0$.



[*Understanding the topology and the geometry of the persistence diagram space via optimal partial transport*, Divol, Lacombe, JACT, 2020]

Persistence image

[*Persistence Images: A Stable Vector Representation of Persistent Homology*, Adams et al., JMLR, 2017]

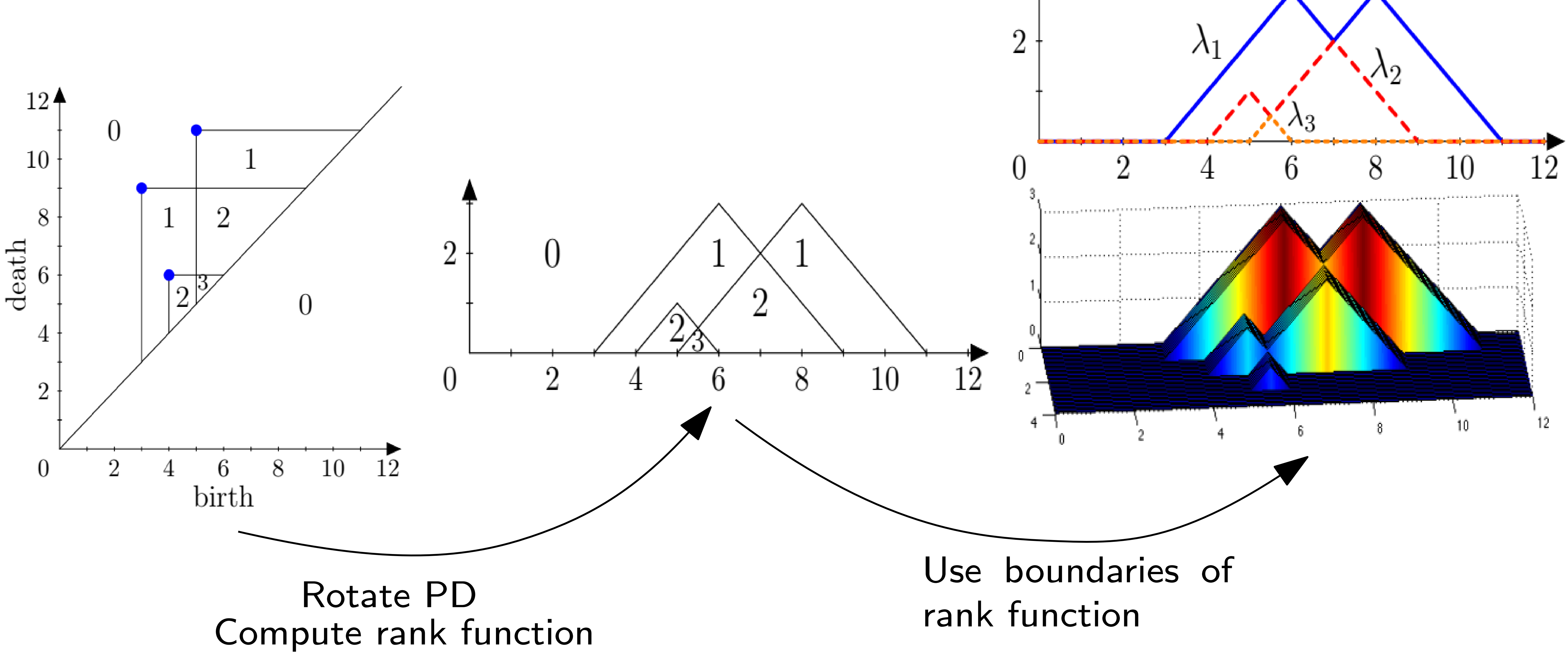


Prop: The following inequalities hold:

- $\|\mathrm{PI}(D) - \mathrm{PI}(D')\|_\infty \leq C(w, \phi_p)\, d_1(D, D')$.
- $\|\mathrm{PI}(D) - \mathrm{PI}(D')\|_2 \leq \sqrt{d} \cdot C(w, \phi_p)\, d_1(D, D')$.

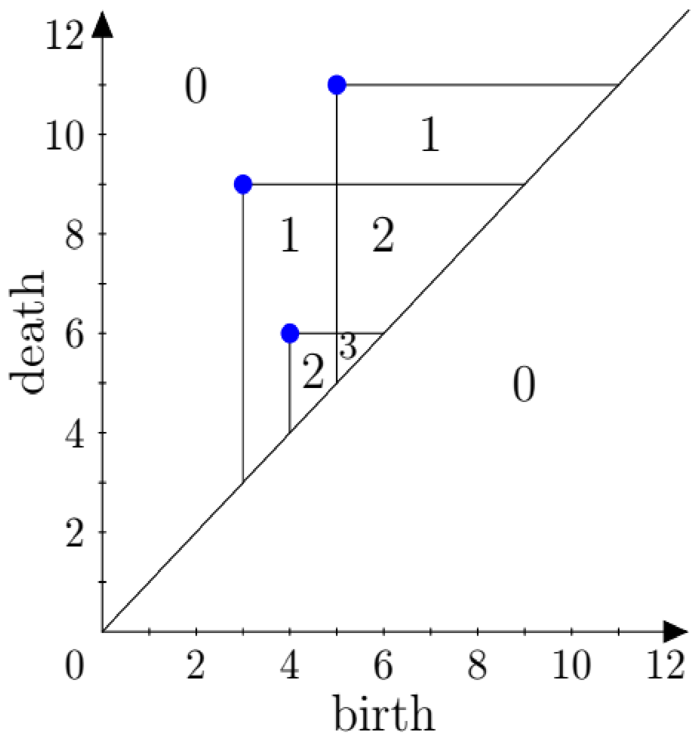
Persistence landscape

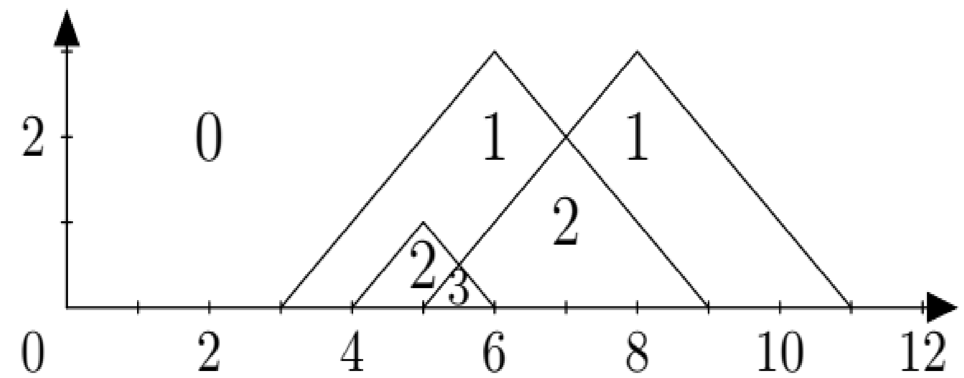
[*Statistical Topological Data Analysis using Persistence Landscapes*, Bubenik, JMLR, 2015]

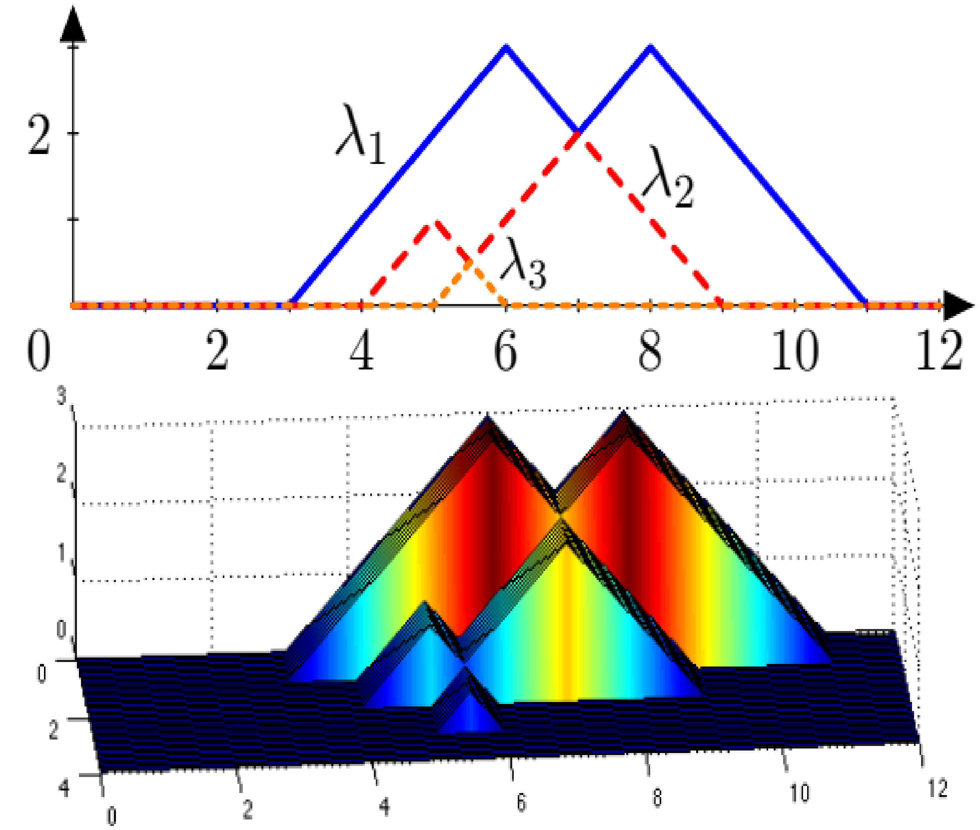


Persistence landscape

[*Statistical Topological Data Analysis using Persistence Landscapes*, Bubenik, JMLR, 2015]







Rotate PD
Compute rank function

Use boundaries of rank function

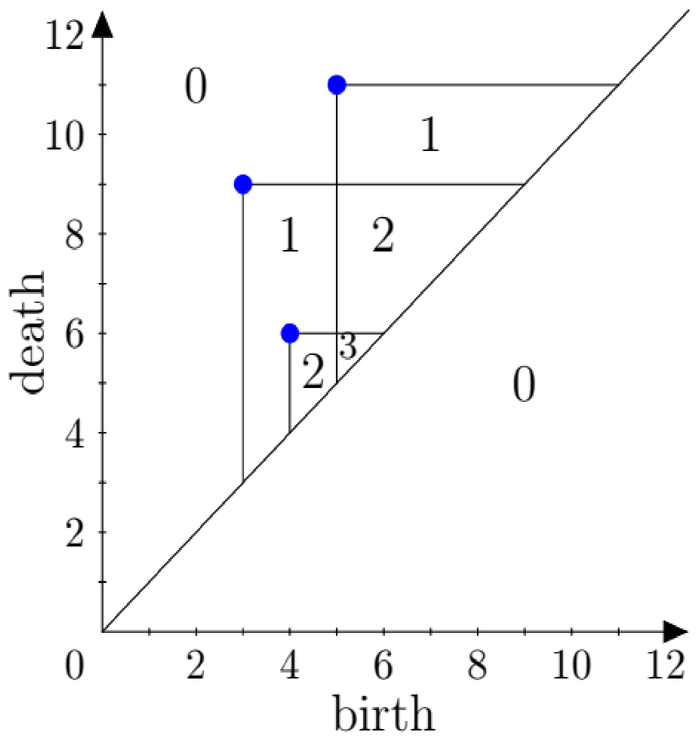
$$x \leq y \Longrightarrow f^{-1}(-\infty, x) \subseteq f^{-1}(-\infty, y)$$

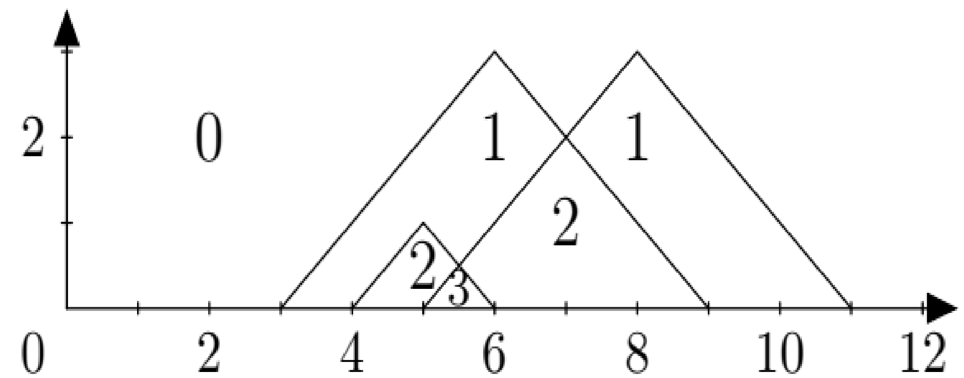
$\iota_x^y : H(f^{-1}(-\infty, x)) \to H(f^{-1}(-\infty, y))$ induced linear map

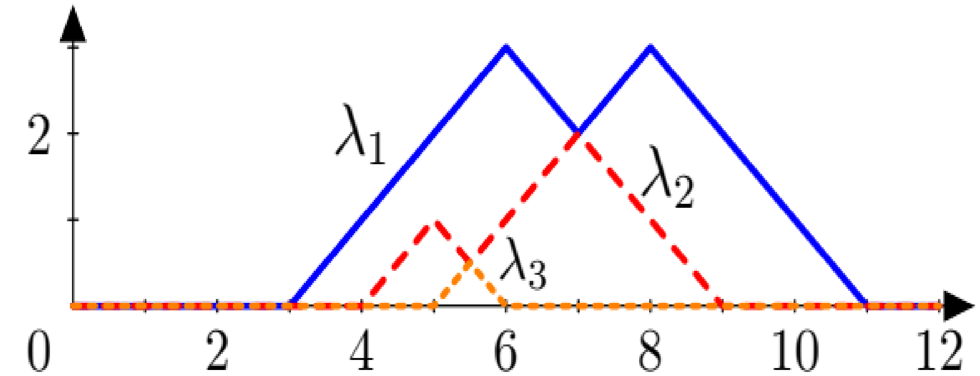
Rank function is defined as $\lambda(x, y) = \operatorname{rank} \iota_x^y$

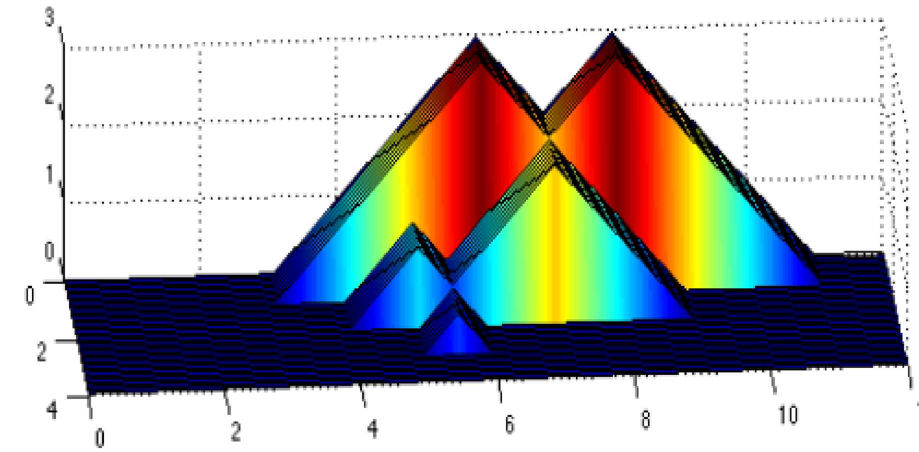
Persistence landscape

[*Statistical Topological Data Analysis using Persistence Landscapes*, Bubenik, JMLR, 2015]









Rotate PD
Compute rank function

Use boundaries of rank function

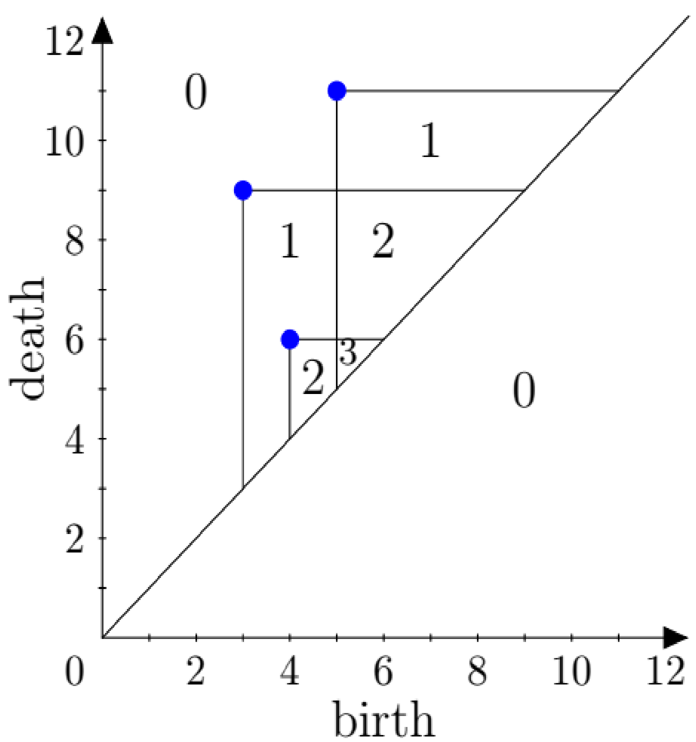
Boundaries of rank function: $\lambda_i(t) = \sup\{s \geq 0 : \lambda(t-s, t+s) \geq i\}$

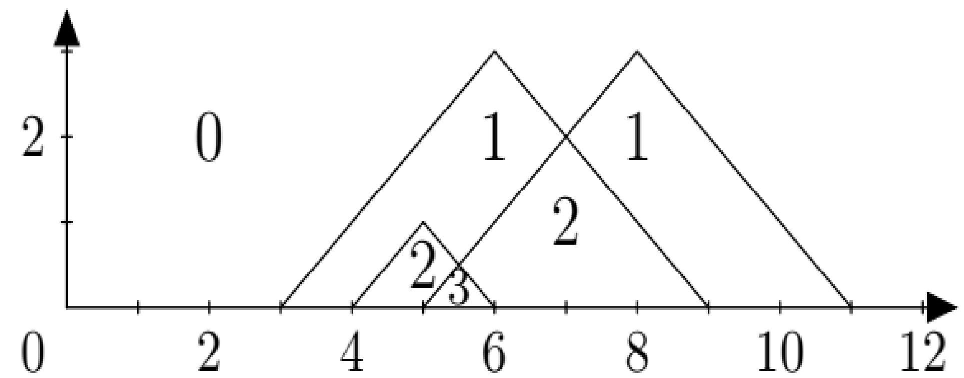
Landscape $\Lambda : \mathbb{R}^2 \to \mathbb{R}$ is defined as: $\Lambda(i,t) = \lambda_{\lfloor i \rfloor}(t)$

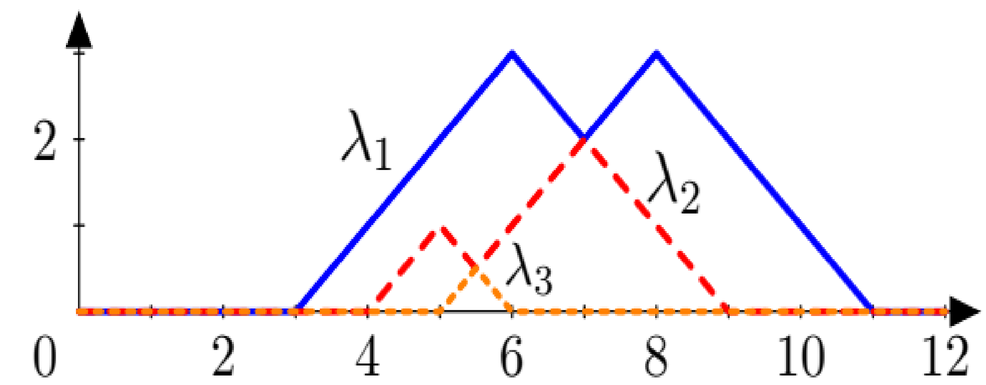
They can equivalently be defined as: $\Lambda(i,t) = i\text{-th} \max\{\lambda_j(t)\}$

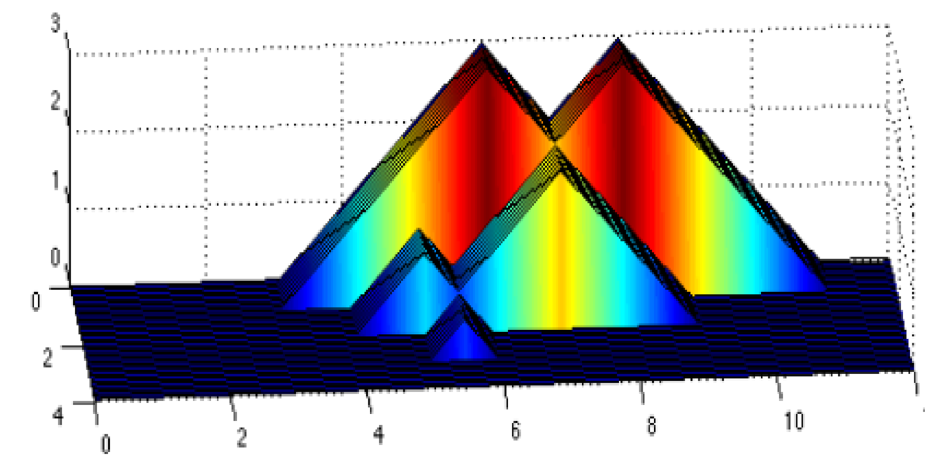
Persistence landscape

[*Statistical Topological Data Analysis using Persistence Landscapes*, Bubenik, JMLR, 2015]









Rotate PD
Compute rank function

Use boundaries of rank function

Prop: The following inequalities hold:

- $\|\Lambda(D) - \Lambda(D')\|_\infty \leq d_b(D, D')$.
- $\min\{1, C(D, D')\|\Lambda(D) - \Lambda(D')\|_2\} \leq d_2(D, D')$.

The Deep Set architecture

[*Deep Sets*, Zaheer, Kottur, Ravanbakhsh, Poczos, Salakhutdinov, Smola, NeurIPS, 2017]

Deep Set is a novel neural net architecture that is able to handle sets instead of finite dimensional vectors

Input: $\{x_1, ..., x_n\} \subset \mathbb{R}^d$ instead of $x \in \mathbb{R}^d$

The Deep Set architecture

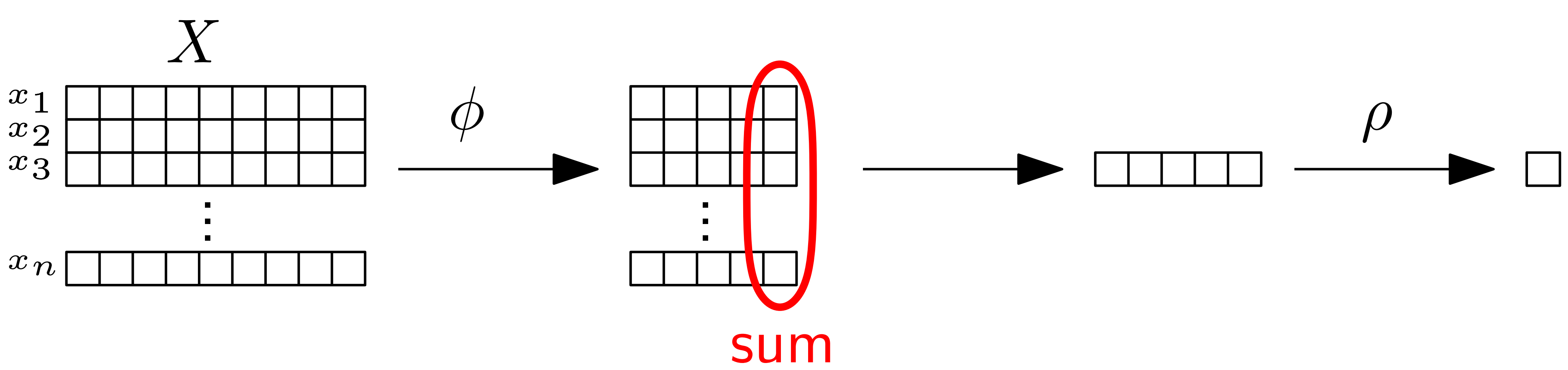
[*Deep Sets*, Zaheer, Kottur, Ravanbakhsh, Poczos, Salakhutdinov, Smola, NeurIPS, 2017]

Deep Set is a novel neural net architecture that is able to handle sets instead of finite dimensional vectors

Input: $\{x_1, ..., x_n\} \subset \mathbb{R}^d$ instead of $x \in \mathbb{R}^d$

Network is *permutation invariant*: $F(X) = \rho\left(\sum_i \phi(x_i)\right)$

$$\Rightarrow F(\{x_1, ..., x_n\}) = F(\{x_{\sigma(1)}, ..., x_{\sigma(n)}\}),\ \forall\sigma$$



In practice: $\phi(x_i) = W \cdot x_i + b$

The Deep Set architecture

[*Deep Sets*, Zaheer, Kottur, Ravanbakhsh, Poczos, Salakhutdinov, Smola, NeurIPS, 2017]

Deep Set is a novel neural net architecture that is able to handle sets instead of finite dimensional vectors

Input: $\{x_1, ..., x_n\} \subset \mathbb{R}^d$ instead of $x \in \mathbb{R}^d$

Network is *permutation invariant*: $F(X) = \rho\left(\sum_i \phi(x_i)\right)$

Universality theorem

Thm: A function f is permutation invariant iif $f(X) = \rho\left(\sum_i \phi(x_i)\right)$ for some ρ and ϕ, whenever X is included in a *countable* space.

Application to PDs

Application to PDs

Permutation invariant layers generalize several TDA approaches

Application to PDs

Permutation invariant layers generalize several TDA approaches

$\rightarrow$ persistence images

Application to PDs

Permutation invariant layers generalize several TDA approaches

$\longrightarrow$ persistence images $\quad \longrightarrow$ landscapes

Application to PDs

Permutation invariant layers generalize several TDA approaches

$\longrightarrow$ persistence images $\longrightarrow$ landscapes $\longrightarrow$ Betti curves [*Time Series Classification via Topological Data Analysis*, Umeda, Trans. Jap. Soc. for AI, 2017]

Application to PDs

Permutation invariant layers generalize several TDA approaches

$\rightarrow$ persistence images $\rightarrow$ landscapes $\rightarrow$ Betti curves [*Time Series Classification via Topological Data Analysis*, Umeda, Trans. Jap. Soc. for AI, 2017]

But not all of them since $\mathbb{R}^2$ is not countable

Application to PDs

Permutation invariant layers generalize several TDA approaches

$\rightarrow$ persistence images $\rightarrow$ landscapes $\rightarrow$ Betti curves [*Time Series Classification via Topological Data Analysis*, Umeda, Trans. Jap. Soc. for AI, 2017]

But not all of them since $\mathbb{R}^2$ is not countable

Using any permutation invariant operation (such as max, min, kth largest value) allows to generalize other TDA approaches

Application to PDs

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]

Permutation invariant layers generalize several TDA approaches

$\rightarrow$ persistence images $\rightarrow$ landscapes $\rightarrow$ Betti curves

[*Time Series Classification via Topological Data Analysis*, Umeda, Trans. Jap. Soc. for AI, 2017]

But not all of them since $\mathbb{R}^2$ is not countable

Using any permutation invariant operation (such as max, min, kth largest value) allows to generalize other TDA approaches

$$\text{PersLay}(D) = \rho\left(\mathsf{op}\{w(p) \cdot \phi(p)\}_{p \in D}\right)$$

Application to PDs

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]

Permutation invariant layers generalize several TDA approaches

$\rightarrow$ persistence images $\rightarrow$ landscapes $\rightarrow$ Betti curves

[*Time Series Classification via Topological Data Analysis*, Umeda, Trans. Jap. Soc. for AI, 2017]

But not all of them since $\mathbb{R}^2$ is not countable

Using any permutation invariant operation (such as max, min, kth largest value) allows to generalize other TDA approaches

$$\mathrm{PersLay}(D) = \rho\left(\mathsf{op}\{w(p) \cdot \phi(p)\}_{p \in D}\right)$$

Permutation-invariant operation

Application to PDs

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]

Permutation invariant layers generalize several TDA approaches

$\rightarrow$ persistence images $\rightarrow$ landscapes $\rightarrow$ Betti curves

[*Time Series Classification via Topological Data Analysis*, Umeda, Trans. Jap. Soc. for AI, 2017]

But not all of them since $\mathbb{R}^2$ is not countable

Using any permutation invariant operation (such as max, min, kth largest value) allows to generalize other TDA approaches

$$\mathrm{PersLay}(D) = \rho\left(\mathsf{op}\{w(p)\cdot\phi(p)\}_{p\in D}\right)$$

Permutation-invariant operation

Weight function

Application to PDs

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]

Permutation invariant layers generalize several TDA approaches

$\rightarrow$ persistence images $\rightarrow$ landscapes $\rightarrow$ Betti curves

[*Time Series Classification via Topological Data Analysis*, Umeda, Trans. Jap. Soc. for AI, 2017]

But not all of them since $\mathbb{R}^2$ is not countable

Using any permutation invariant operation (such as max, min, kth largest value) allows to generalize other TDA approaches

$$\mathrm{PersLay}(D) = \rho\left(\mathsf{op}\{w(p)\cdot\phi(p)\}_{p\in D}\right)$$

Permutation-invariant operation

Weight function

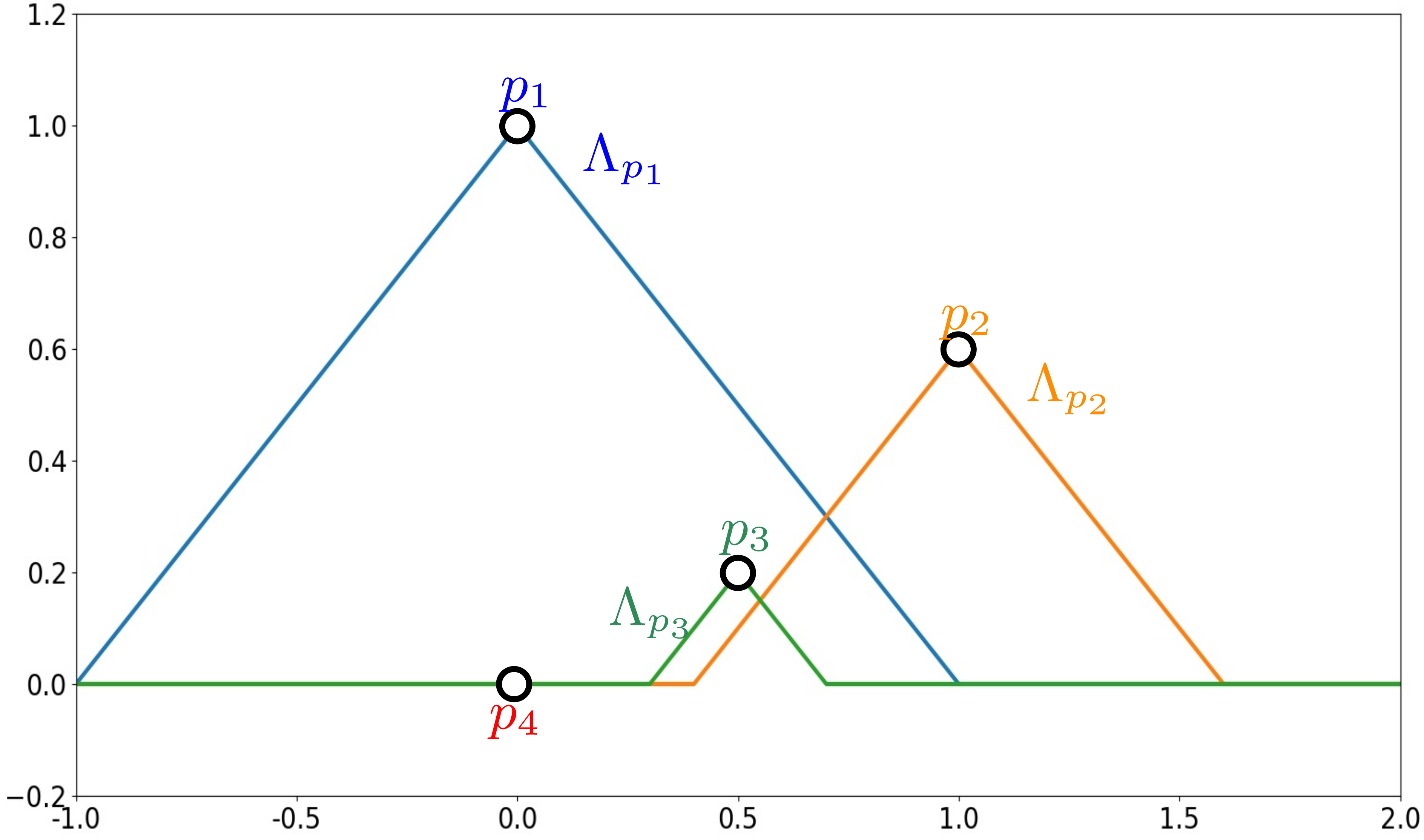
Point transformation

Application to PDs

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]

Parameters $t_1, \cdots, t_q \in \mathbb{R}$

$$w(p) = 1 \qquad \phi_\Lambda : p \mapsto \begin{bmatrix} \Lambda_p(t_1) \\ \Lambda_p(t_2) \\ \vdots \\ \Lambda_p(t_q) \end{bmatrix} \qquad \text{op} = \text{top-}k$$

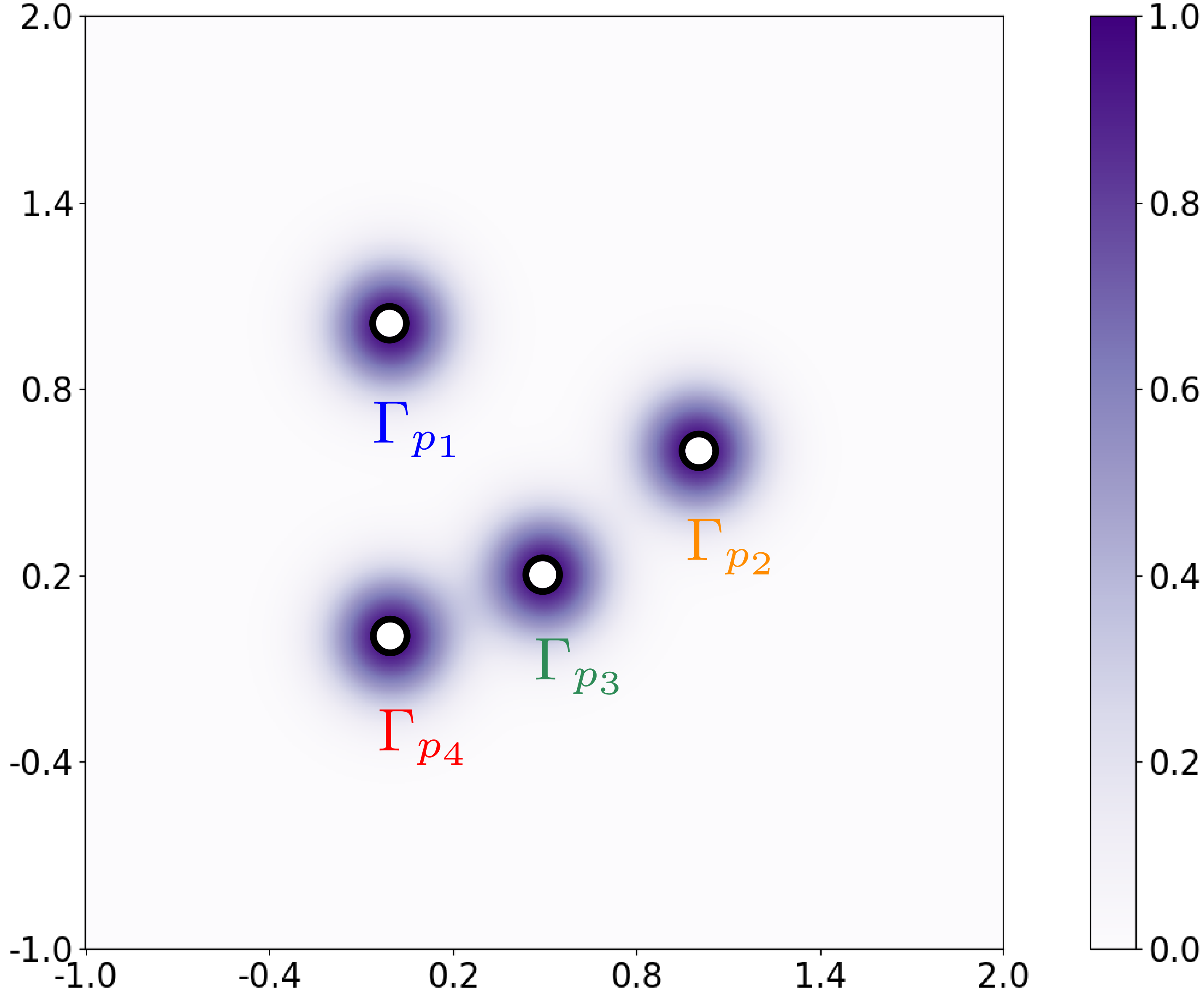


Application to PDs

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]

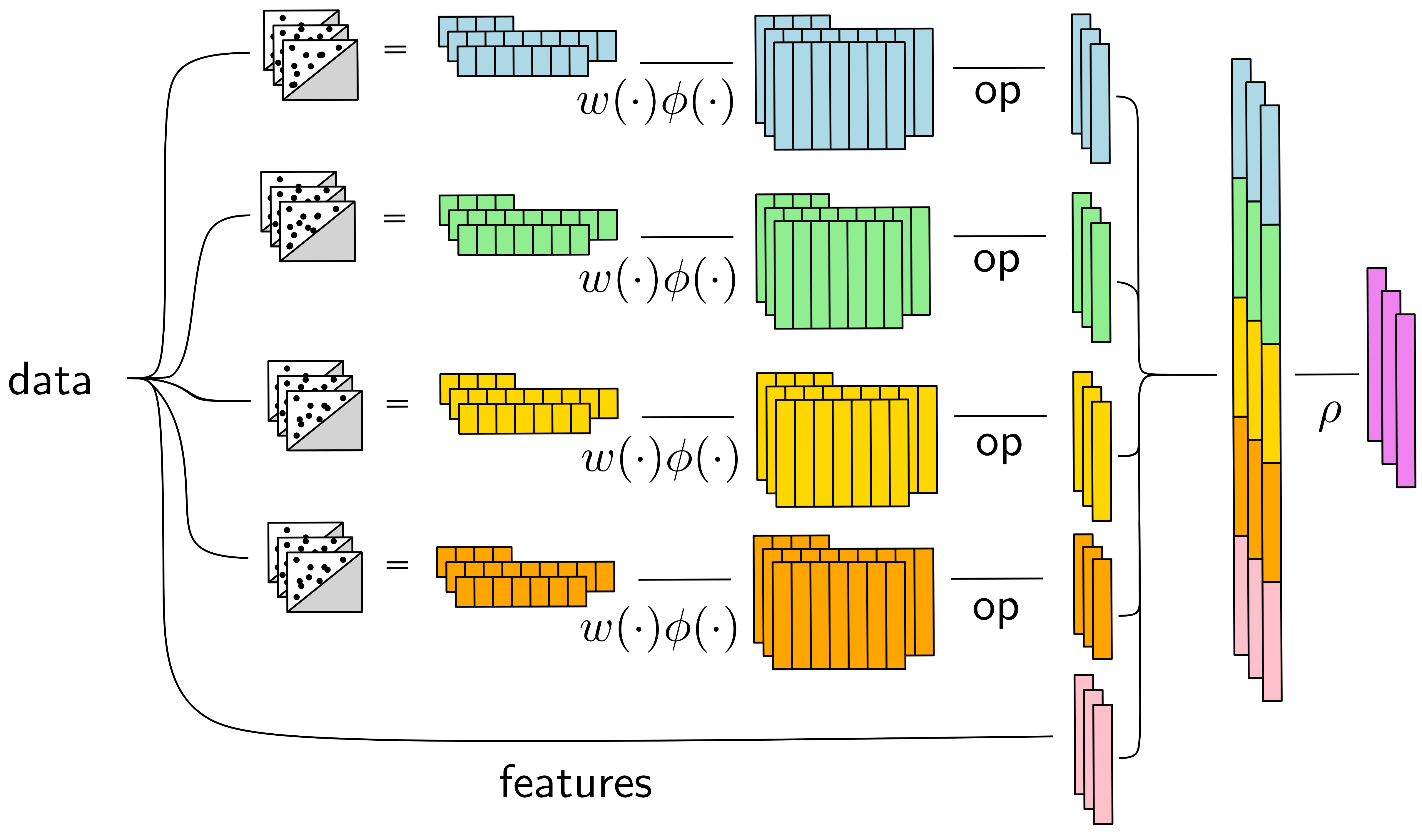
Parameters $t_1, \cdots, t_q \in \mathbb{R}^2$

$$w(p) = w_t((x, y)) \qquad \phi_\Gamma : p \mapsto \begin{bmatrix} \Gamma_p(t_1) \\ \Gamma_p(t_2) \\ \vdots \\ \Gamma_p(t_q) \end{bmatrix} \qquad \text{op} = \text{sum}$$



Application to PDs

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]



Application to graph classification

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]

Let $G = (V, E)$ be a graph, A its adjacency matrix

D its degree matrix

and $L_w(G) = I - D^{-1/2} A D^{-1/2}$ its normalized Laplacian.

Application to graph classification

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]

Let $G = (V, E)$ be a graph, A its adjacency matrix

D its degree matrix

and $L_w(G) = I - D^{-1/2} A D^{-1/2}$ its normalized Laplacian.

$L_w(G)$ decomposes on a orthonormal basis $\phi_1 \dots \phi_n$

with eigenvalues $0 \leq \lambda_1 \leq \cdots \leq \lambda_n \leq 2$

Application to graph classification

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]

Let $G = (V, E)$ be a graph, A its adjacency matrix

D its degree matrix

and $L_w(G) = I - D^{-1/2}AD^{-1/2}$ its normalized Laplacian.

$L_w(G)$ decomposes on a orthonormal basis $\phi_1 \dots \phi_n$

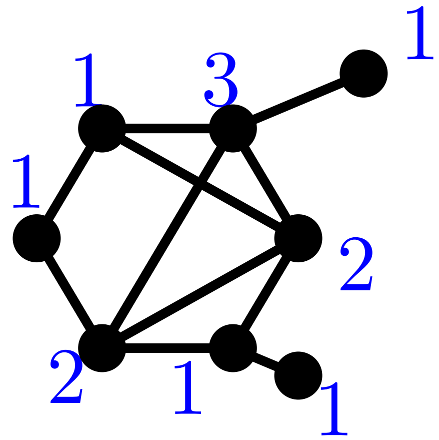
with eigenvalues $0 \leq \lambda_1 \leq \cdots \leq \lambda_n \leq 2$

Def: Let $t \geq 0$, and define the *Heat Kernel Signature* of param t:

$$\mathrm{hks}_{G,t} : v \mapsto \sum_{k=1}^{n} \exp(-\lambda_k t)\phi_k(v)^2$$

Application to graph classification

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]

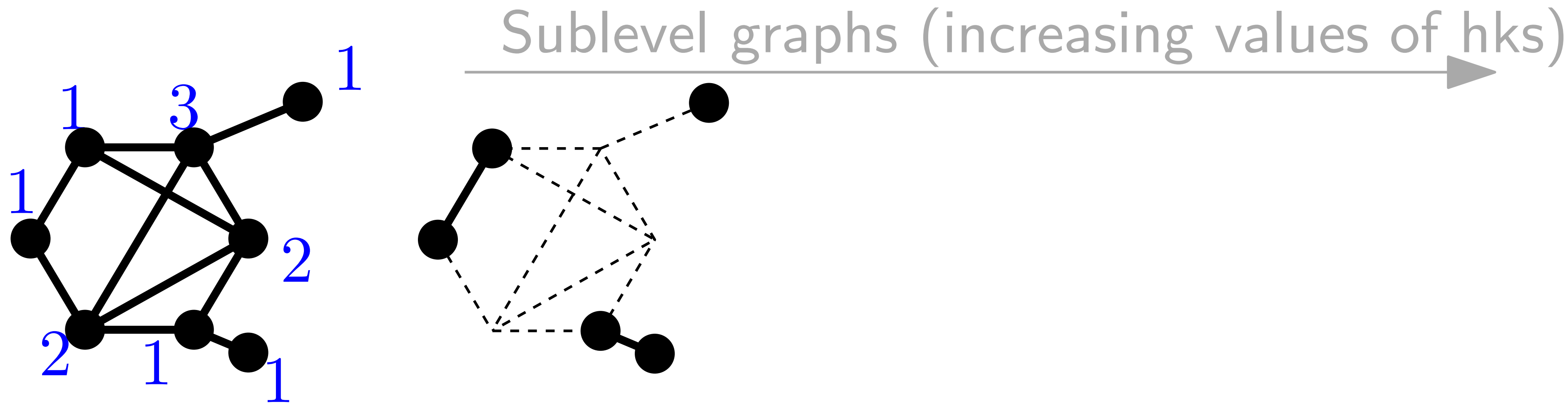


Def: Let $t \geq 0$, and define the *Heat Kernel Signature* of param t:

$$\mathrm{hks}_{G,t} : v \mapsto \sum_{k=1}^{n} \exp(-\lambda_k t)\phi_k(v)^2$$

Application to graph classification

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]

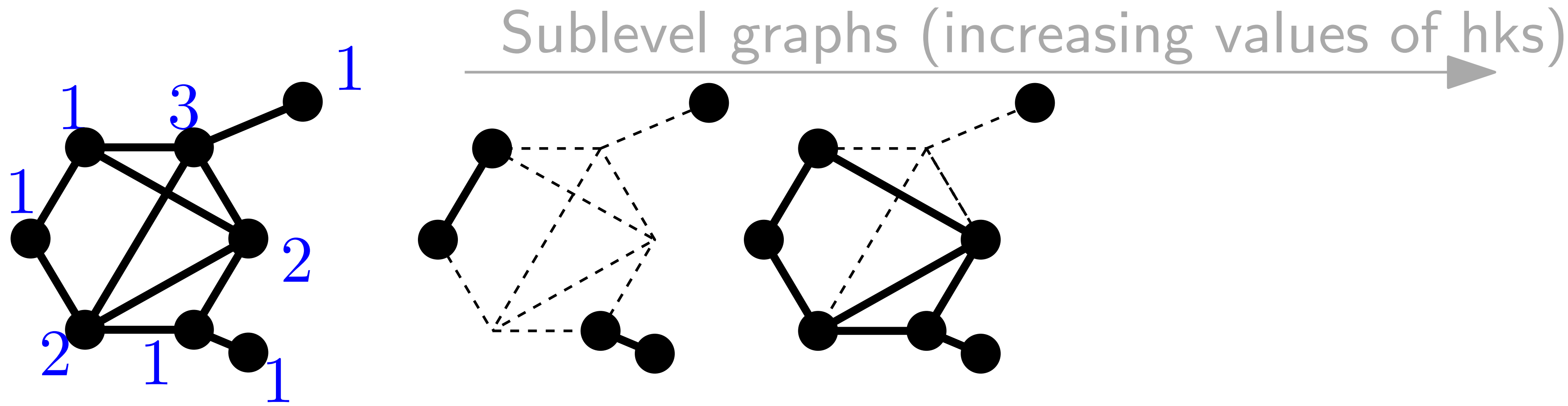


Def: Let $t \geq 0$, and define the *Heat Kernel Signature* of param t:

$$\mathrm{hks}_{G,t} : v \mapsto \sum_{k=1}^{n} \exp(-\lambda_k t)\phi_k(v)^2$$

Application to graph classification

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]

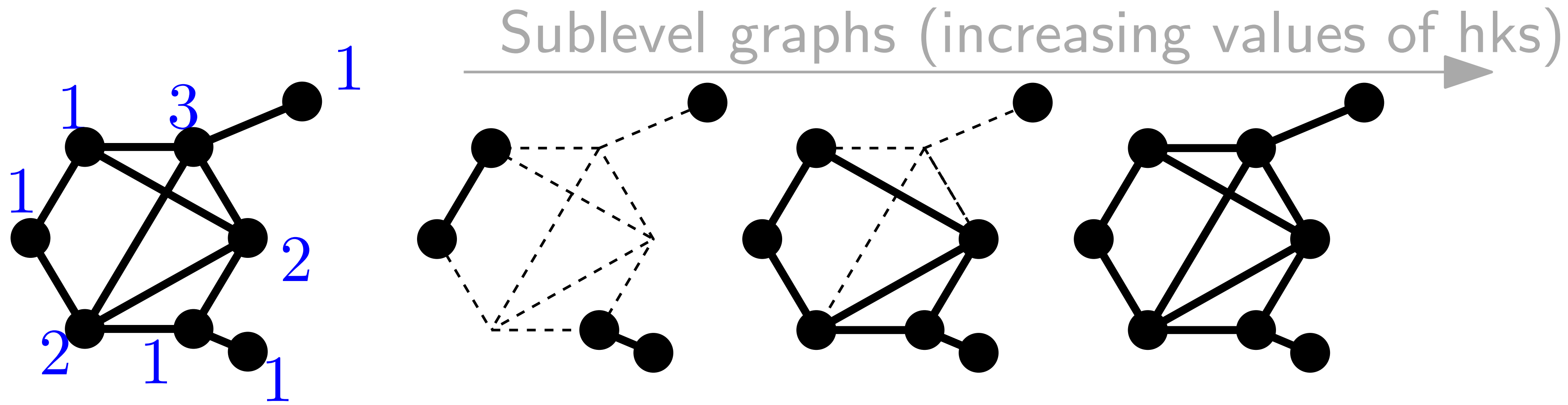


Def: Let $t \geq 0$, and define the *Heat Kernel Signature* of param t:

$$\mathrm{hks}_{G,t} : v \mapsto \sum_{k=1}^{n} \exp(-\lambda_k t)\phi_k(v)^2$$

Application to graph classification

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]

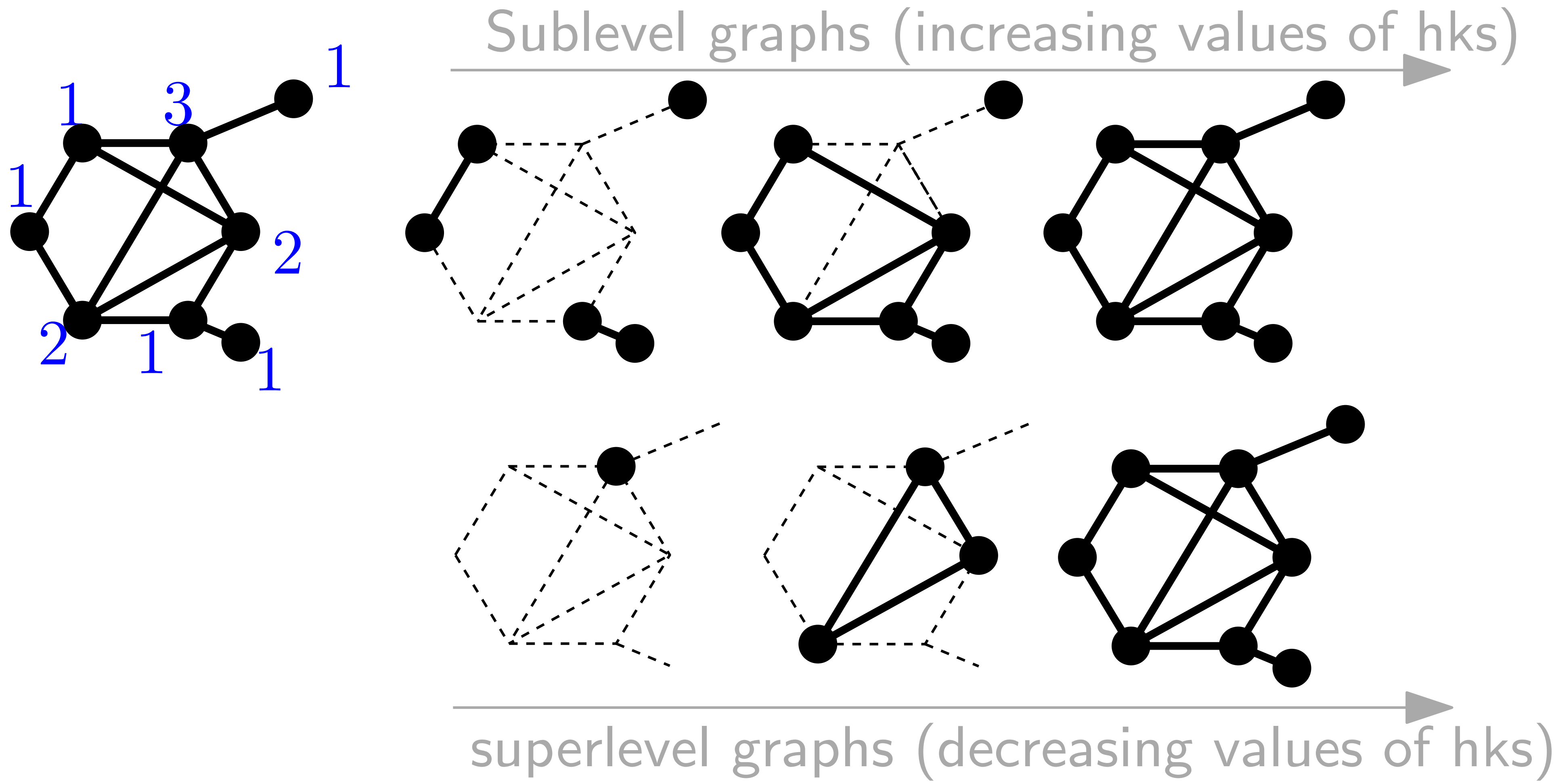


Def: Let $t \geq 0$, and define the *Heat Kernel Signature* of param t:

$$\mathrm{hks}_{G,t} : v \mapsto \sum_{k=1}^{n} \exp(-\lambda_k t)\phi_k(v)^2$$

Application to graph classification

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]



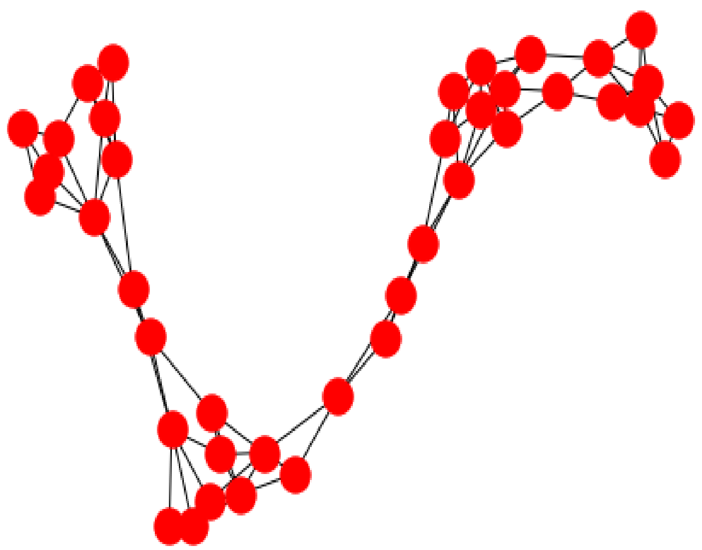
Def: Let $t \geq 0$, and define the *Heat Kernel Signature* of param t:

$$\mathrm{hks}_{G,t} : v \mapsto \sum_{k=1}^{n} \exp(-\lambda_k t)\phi_k(v)^2$$

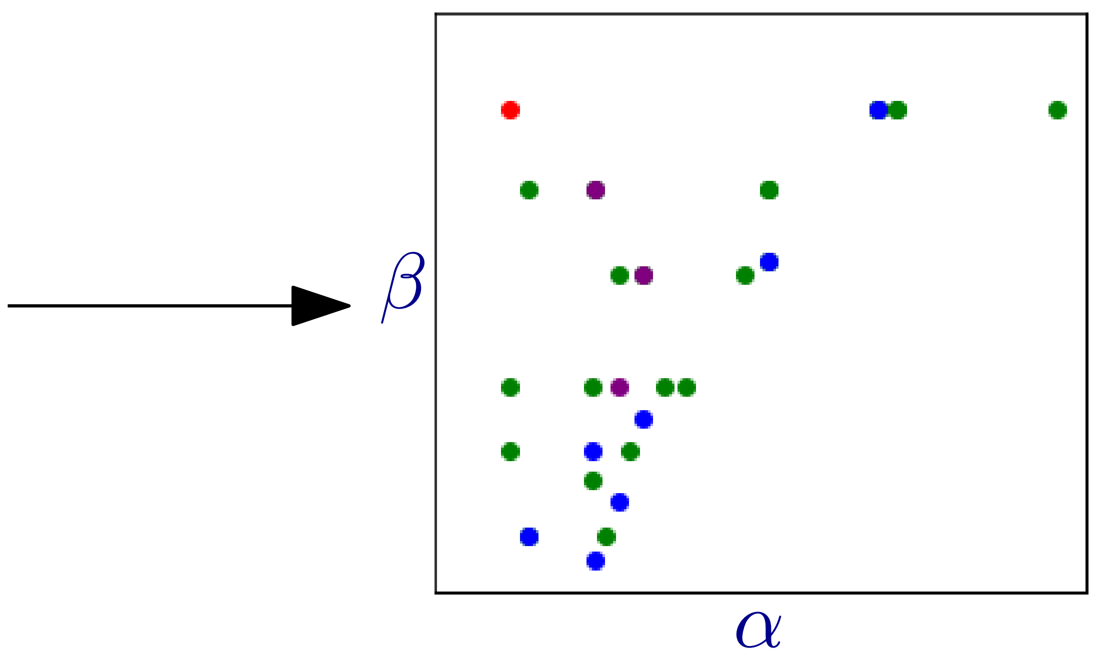
Application to graph classification

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]

Graph from the `PROTEINS` dataset

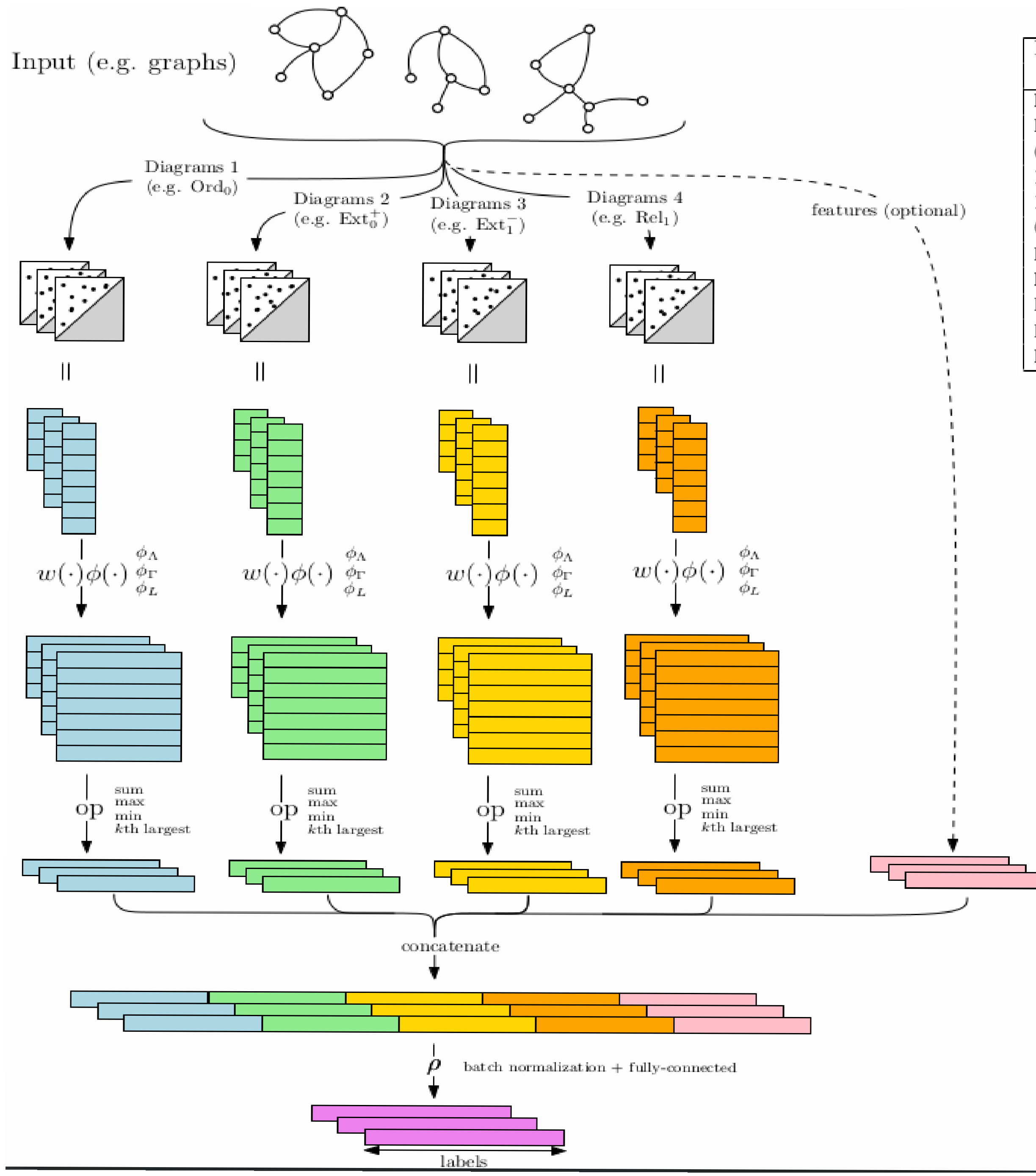


Corresponding persistence diagram



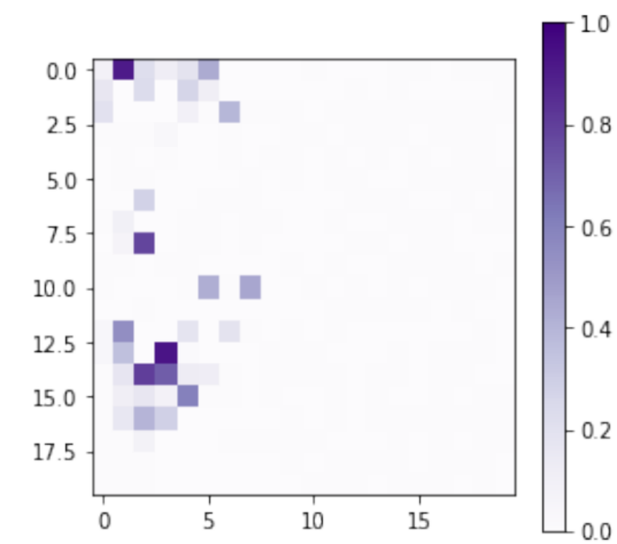
Application to graph classification

[*PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*, C., Chazal, Ike, Lacombe, Royer, Umeda, AISTATS, 2019]



Dataset	SV[1]	RetGK* [2]	FGSD [3]	GCNN [4]	GIN [5]	PersLay	
						Mean	Max
REDDIT5K	—	56.1	47.8	52.9	57.0	55.6	56.5
REDDIT12K	—	48.7	—	46.6	—	47.7	49.1
COLLAB	—	81.0	80.0	79.6	80.1	76.4	78.0
IMDB-B	72.9	71.9	73.6	73.1	74.3	71.2	72.6
IMDB-M	50.3	47.7	52.4	50.3	52.1	48.8	52.2
COX2*	78.4	80.1	—	—	—	80.9	81.6
DHFR*	78.4	81.5	—	—	—	80.3	80.9
MUTAG*	88.3	90.3	92.1	86.7	89.0	89.8	91.5
PROTEINS*	72.6	75.8	73.4	76.3	75.9	74.8	75.9
NCI1*	71.6	84.5	79.8	78.4	82.7	73.5	74.0
NCI109*	70.5	—	78.8	—	—	69.5	70.1

Weight function learnt



(after training on the `MUTAG` dataset)

Application to Pearson correlation

Method:

1. Extract local point clouds corresponding to several measurement spots.

Application to Pearson correlation

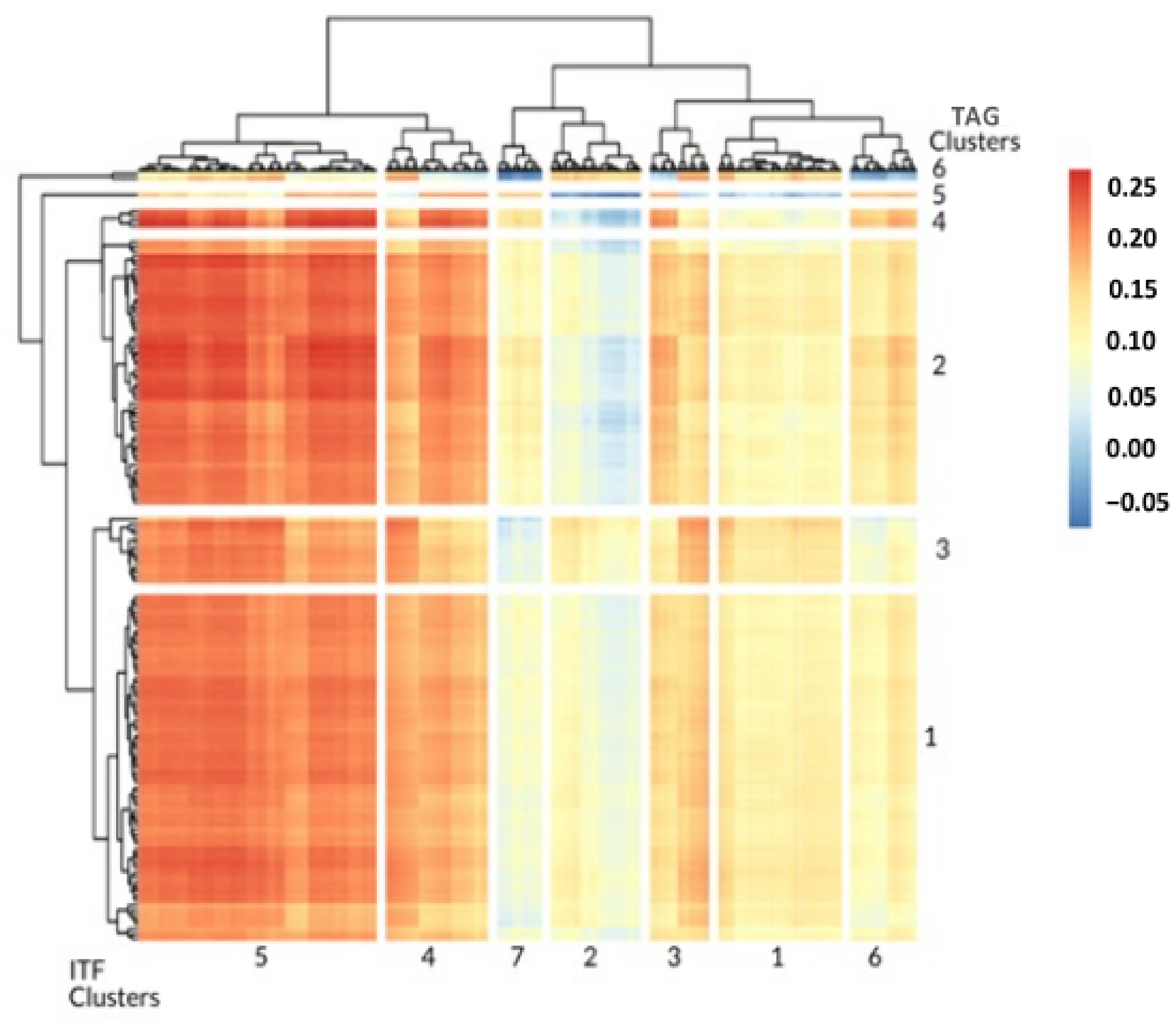
Method:

1. Extract local point clouds corresponding to several measurement spots.
2. Compute persistence images (PIMs) associated to Rips PDs of local point clouds.

Application to Pearson correlation

Method:

1. Extract local point clouds corresponding to several measurement spots.
2. Compute persistence images (PIMs) associated to Rips PDs of local point clouds.
3. Cluster Image Topological Features (ITFs), i.e., PIM pixels, and marker genes, and compute all pairwise correlations.



Application to Pearson correlation

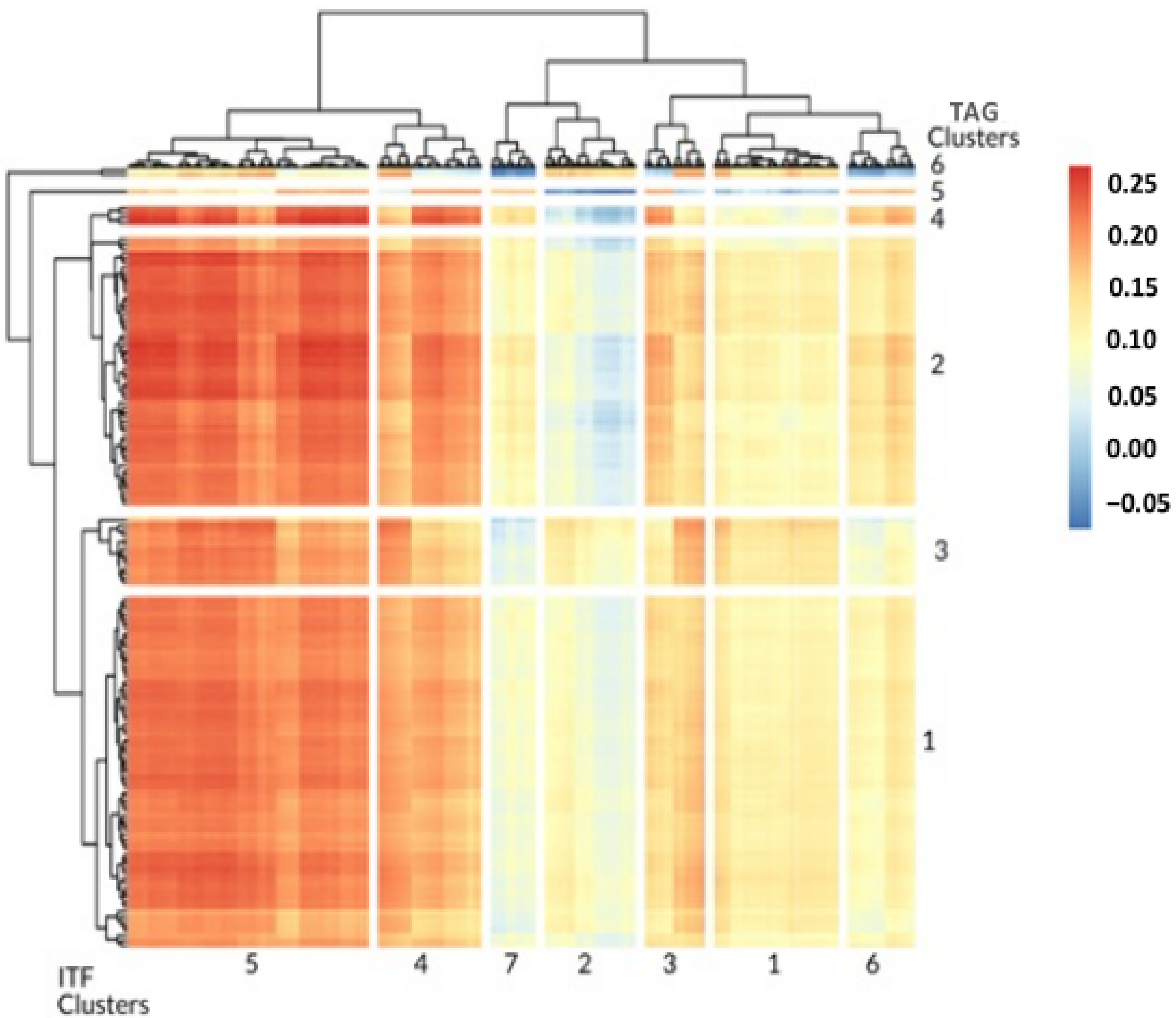
Method:

1. Extract local point clouds corresponding to several measurement spots.

2. Compute persistence images (PIMs) associated to Rips PDs of local point clouds.

3. Cluster Image Topological Features (ITFs), i.e., PIM pixels, and marker genes, and compute all pairwise correlations.

4. Retrieve marker genes with highest correlations and match these *topologically associated genes* (TAGs) against gene ontology.



Term ID	Term name	Adjusted p-value
GO:0005615	Extracellular space	5.57×10^{-5}
GO:0070062	Extracellular exosome	1.27×10^{-3}
GO:1903561	Extracellular vescicle	1.41×10^{-3}
GO:0043230	Extracellular organelle	1.41×10^{-3}

Application to Pearson correlation

Method:

1. Extract local point clouds corresponding to several measurement spots.

2. Compute persistence images (PIMs) associated to Rips PDs of local point clouds.

3. Cluster Image Topological Features (ITFs), i.e., PIM pixels, and marker genes, and compute all pairwise correlations.

4. Retrieve marker genes with highest correlations and match these *topologically associated genes* (TAGs) against gene ontology.

5. Predict TAG expression from ITFs only.