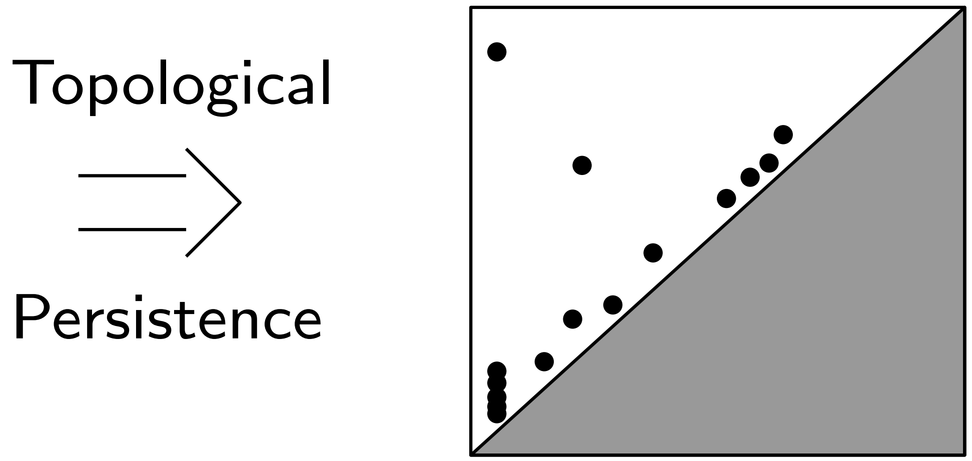
# Topological Data Analysis and Machine Learning

# The TDA pipeline

**Data**



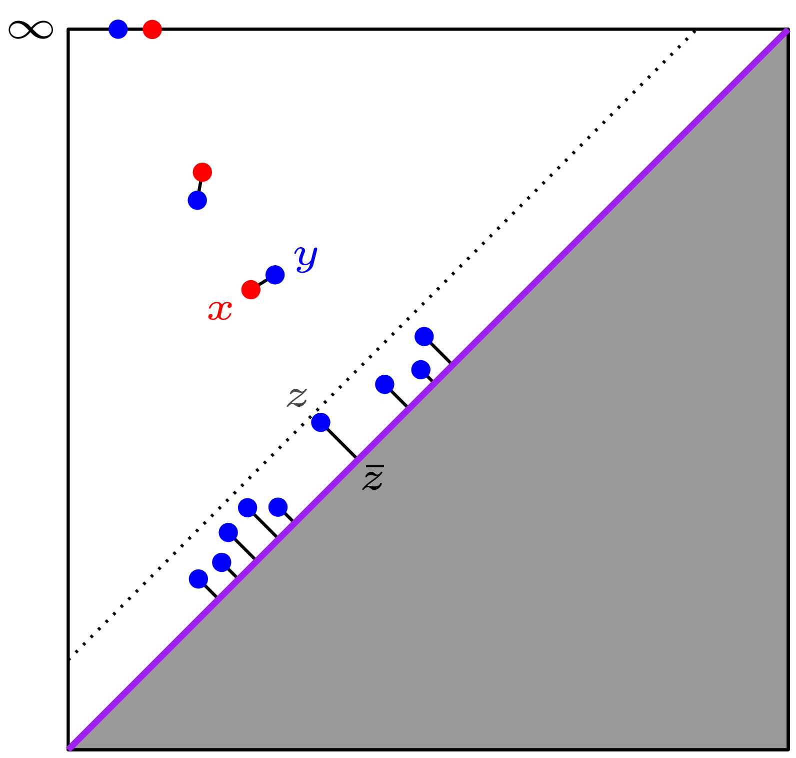

**Descriptors**

**Def:** $p$-th diagram distance (extended metric):

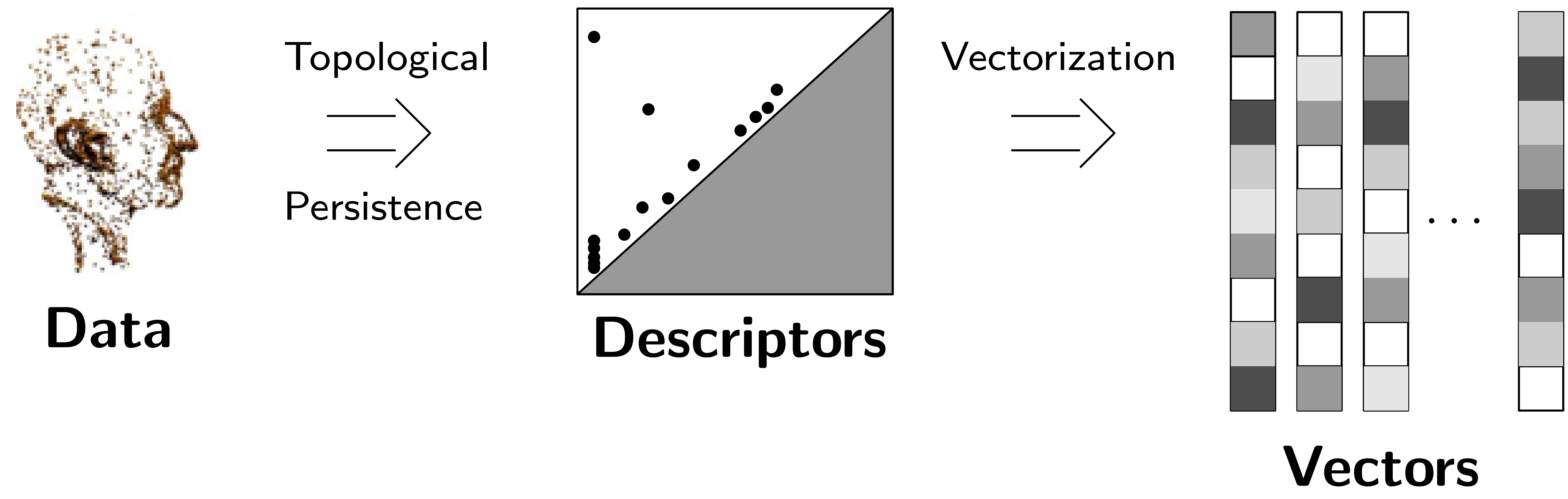
$$\mathrm{d}_p(\mathrm{Dgm}\, f, \mathrm{Dgm}\, g) := \inf_{\Gamma \subseteq \mathrm{Dgm}\, f \times \mathrm{Dgm}\, g} c_p(\Gamma)$$

**Def:** bottleneck distance:

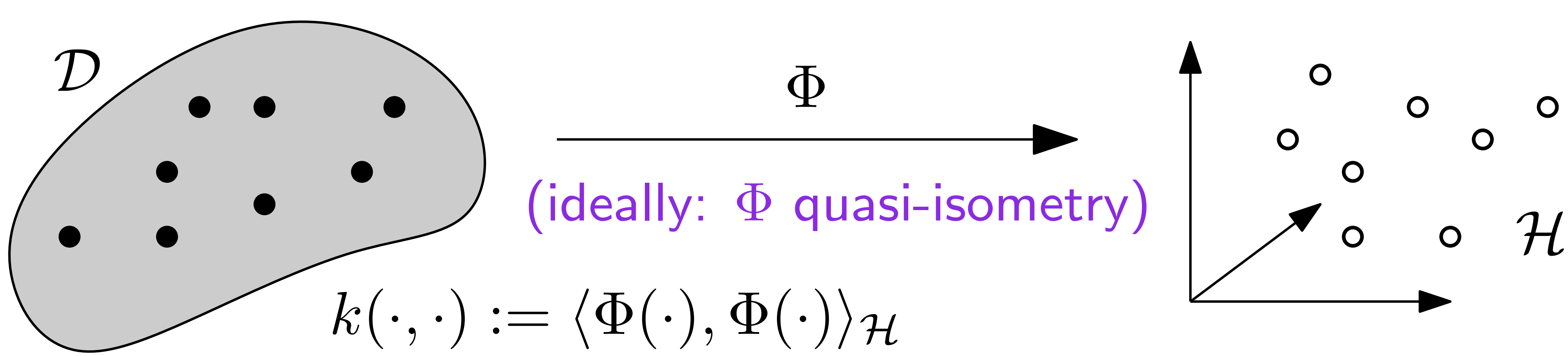
$$\mathrm{d}_\infty(\mathrm{Dgm}\, f, \mathrm{Dgm}\, g) := \lim_{p \to \infty} \mathrm{d}_p(\mathrm{Dgm}\, f, \mathrm{Dgm}\, g)$$



# The TDA pipeline



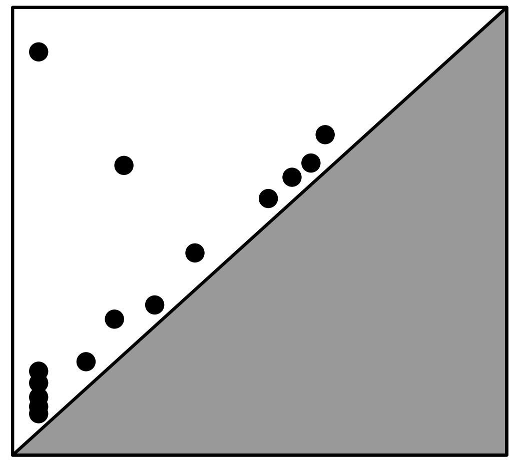

**Vectorization:** map diagrams to (possibly infinite) Hilbert space and use kernel trick



# The TDA pipeline

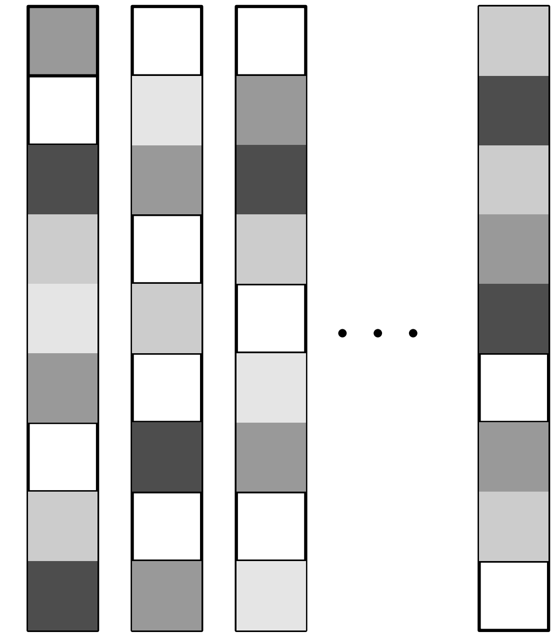
**Data**

Topological ⟹ Persistence



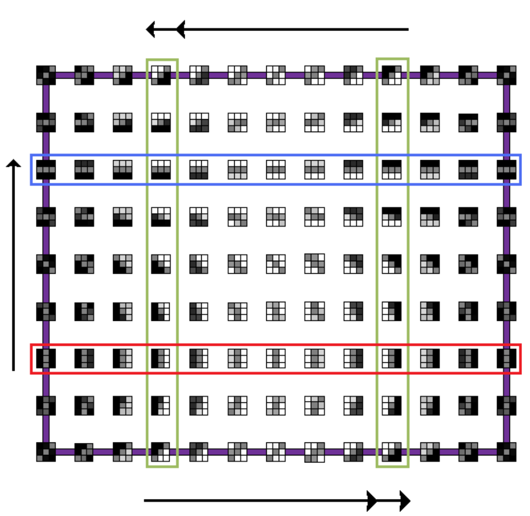
**Descriptors**

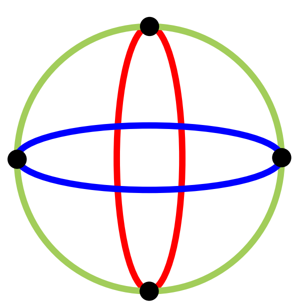
Vectorization ⟹

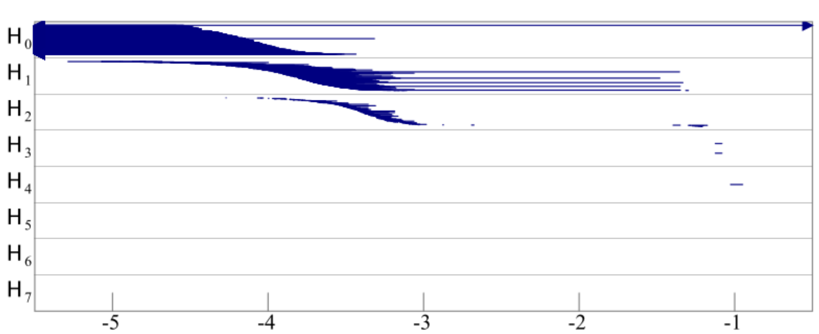


**Vectors**

**Data exploration**

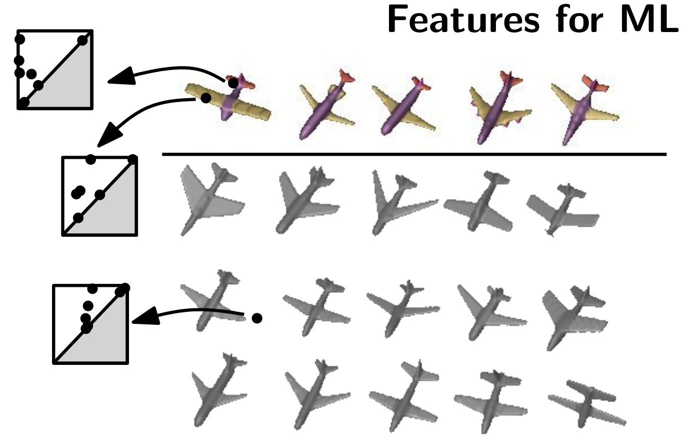






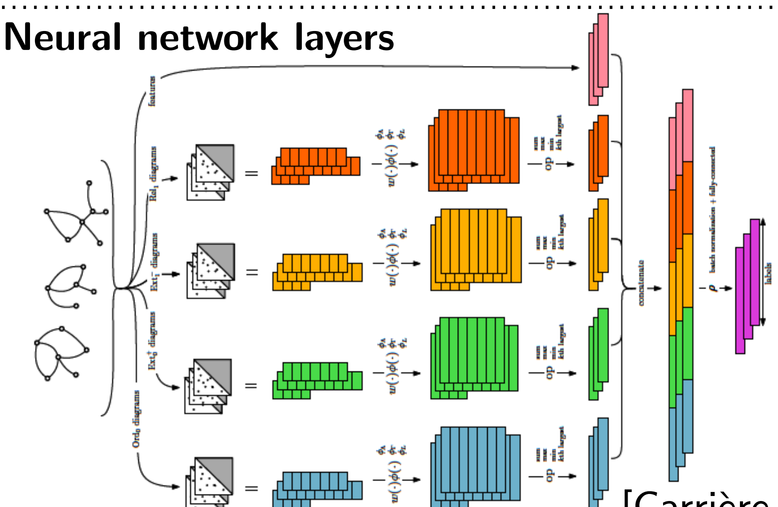

[Ishkhanov et al. '08]

**Features for ML**



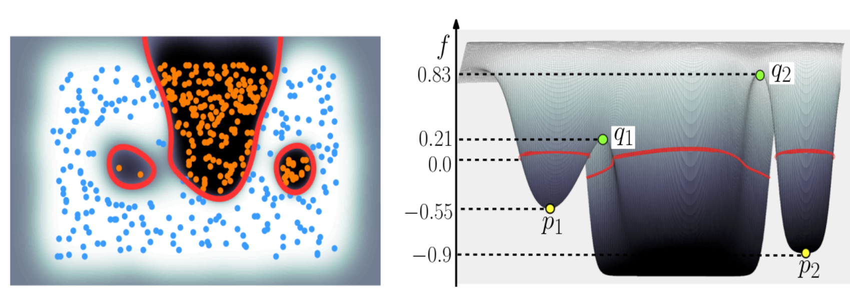
[Carrière et al. '15]

**Neural network layers**



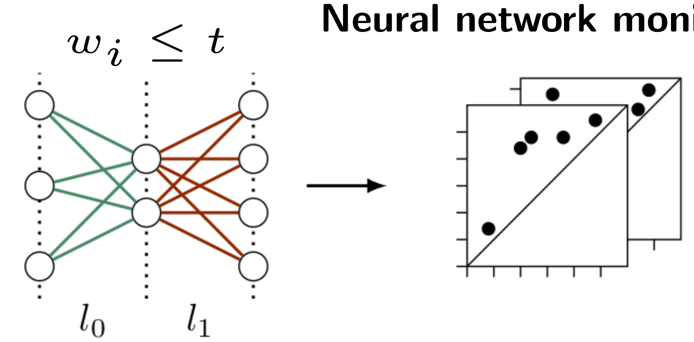
[Carrière et al. '20]

**Topological regularizers**




[Chen et al. '19]

**Neural network monitoring**

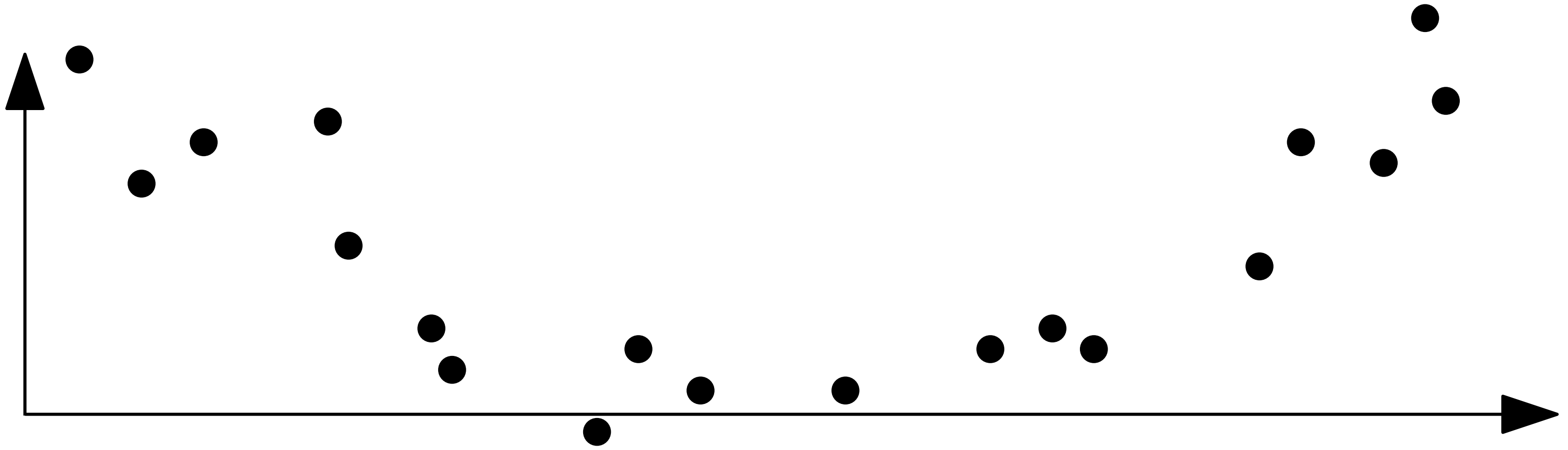



[Rieck et al. '19]

# Detour: Supervised Machine Learning

**Input:** $n$ observations + responses $(x_1, y_1), \cdots, (x_n, y_n) \in X \times Y$

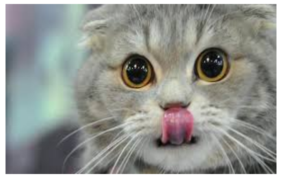
$Y = \mathbb{R}$

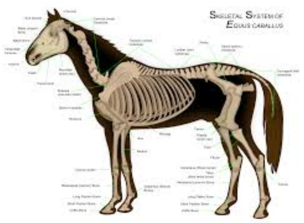


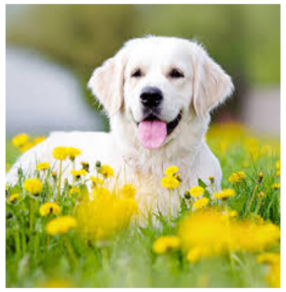
regression

$X = \mathbb{R}$

classification





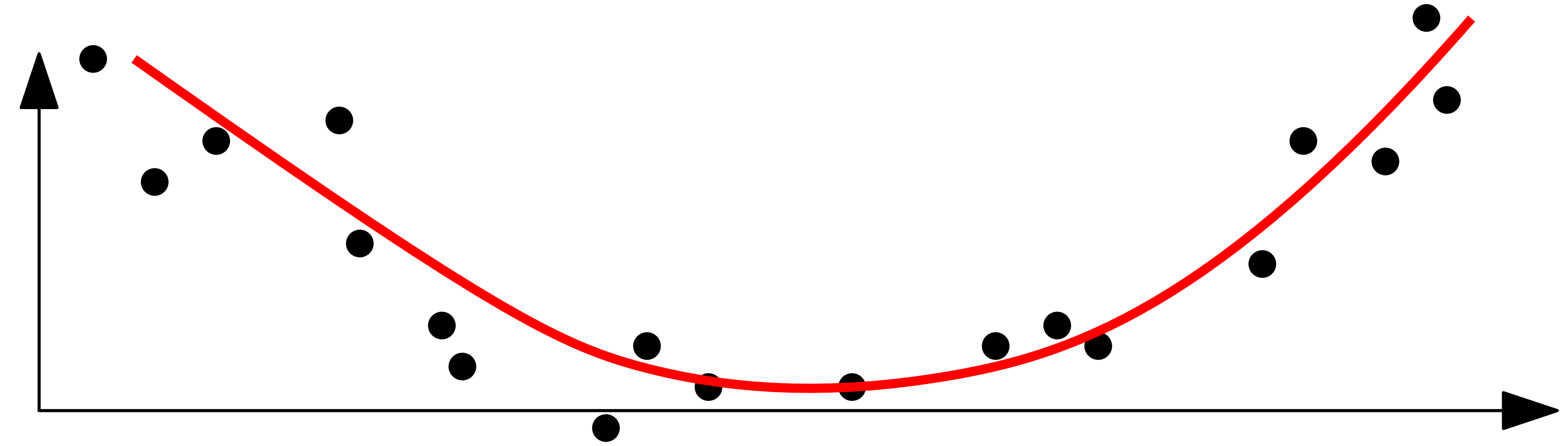


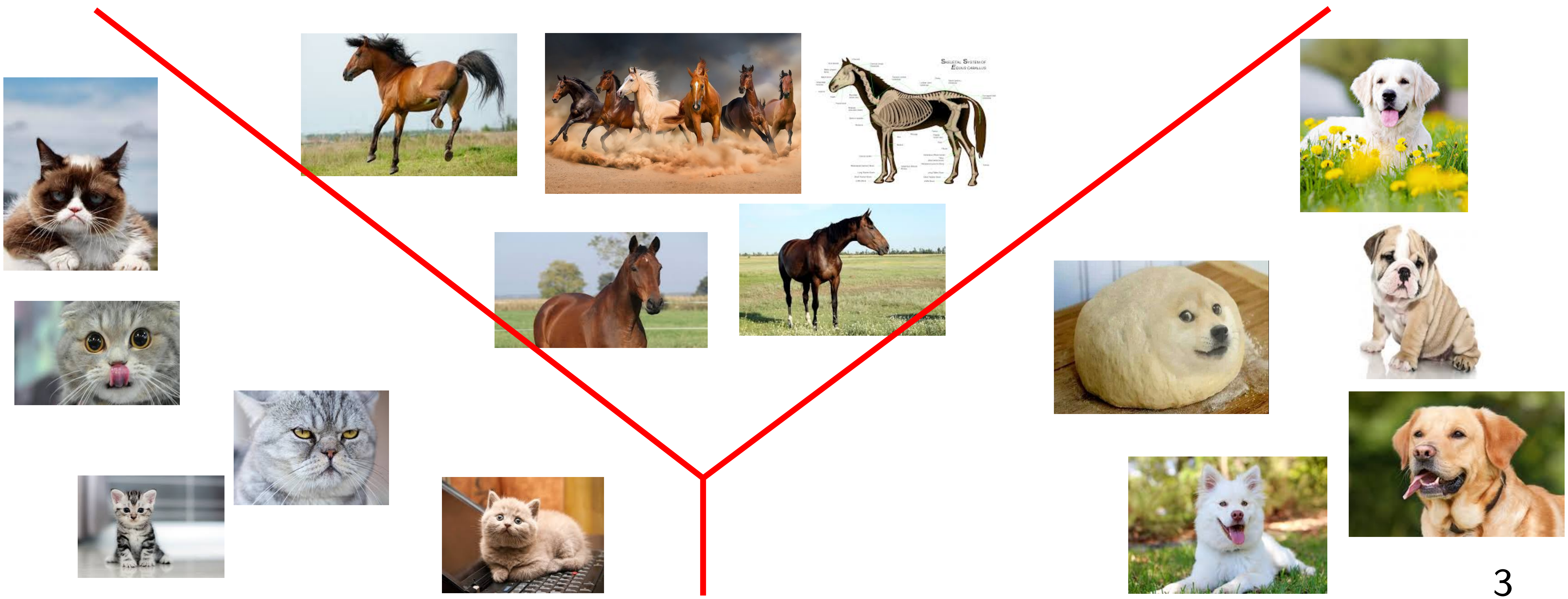
$X =$ images,
$Y = \{$cat, dog, horse$\}$

# Detour: Supervised Machine Learning

**Input:** $n$ observations + responses $(x_1, y_1), \cdots, (x_n, y_n) \in X \times Y$

**Goal:** build a predictor $f : X \to Y$ from $(x_1, y_1), \cdots, (x_n, y_n)$





# Empirical Risk Minimization

Optimization problem (supervised regression / classification):

$$f^* = \arg\min_{f \in \mathcal{F}} \; \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)) + \Omega(f)$$

$\mathcal{F}$ is the **class of predictors**

$L : X \times X \to \mathbb{R}$ is the **loss function**

$\Omega : \mathcal{F} \to \mathbb{R}$ is the **regularizer**

# Empirical Risk Minimization

Optimization problem (supervised regression / classification):

$$f^* = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)) + \Omega(f)$$

$\mathcal{F}$ is the **class of predictors**

$L : X \times X \to \mathbb{R}$ is the **loss function**

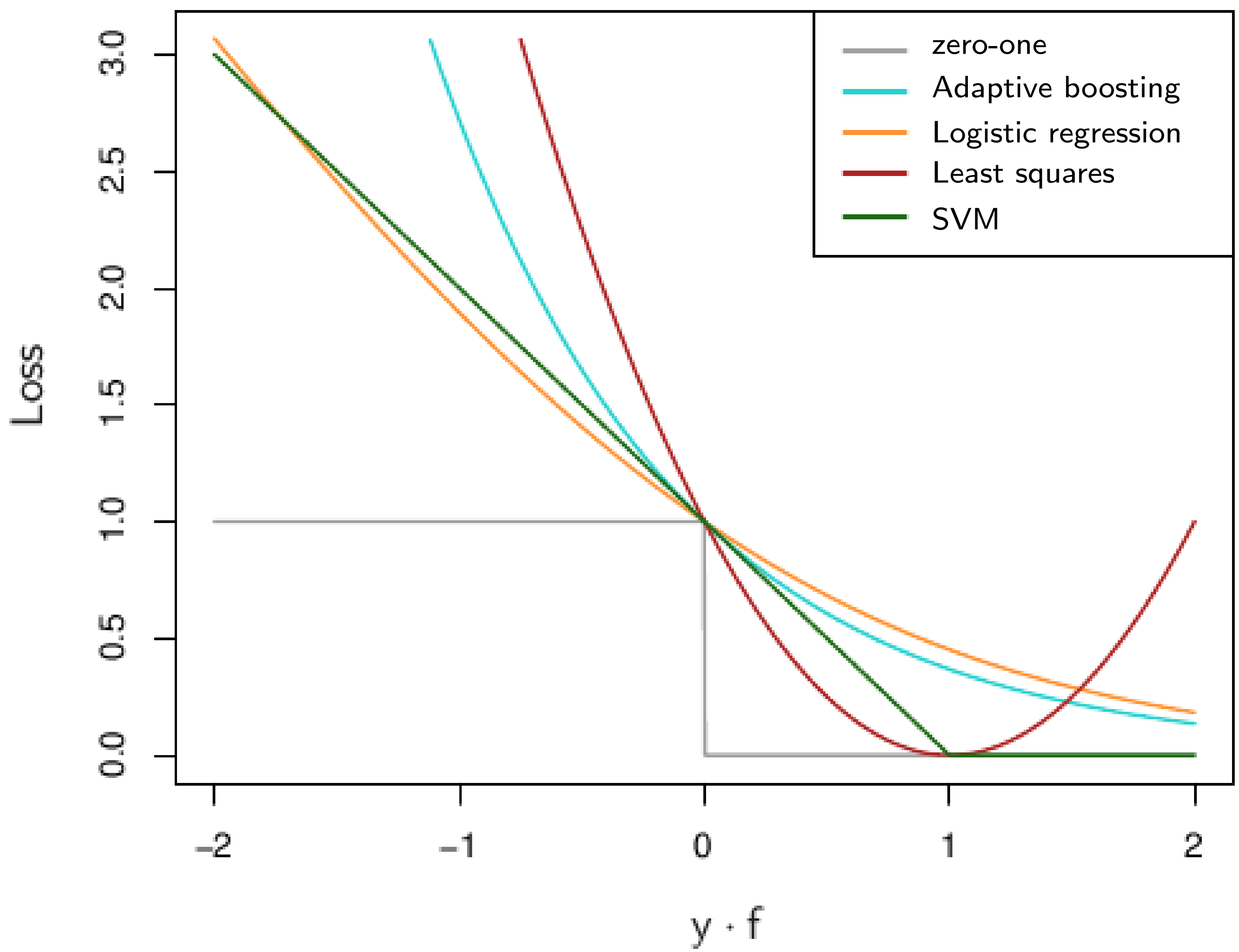
$\Omega : \mathcal{F} \to \mathbb{R}$ is the **regularizer**

| $L(y_i, f(x_i))$ | Name | |
|---|---|---|
| $\mathbb{1}_{y_i \neq f(x_i)}$ | zero-one | $\to$ Bayes |
| $\max\{0, 1 - y_i f(x_i)\}$ | hinge | $\to$ Support Vector Machines |
| $\exp(-y_i f(x_i))$ | exponential | $\to$ Adaptive boosting |
| $\log(1 + \exp(-y_i f(x_i)))$ | logistic | $\to$ Logistic regression |
| $(y_i - f(x_i))^2$ | squared | $\to$ Least squares |

# Empirical Risk Minimization

Optimization problem (supervised regression / classification):

$$f^* = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)) + \Omega(f)$$



# Empirical Risk Minimization

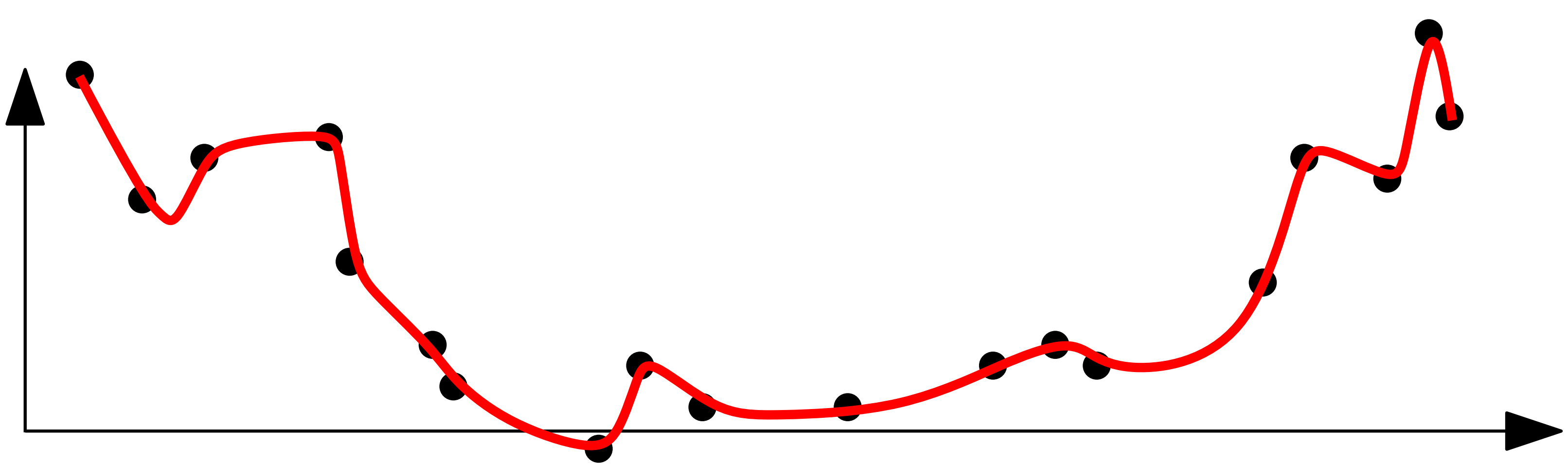
Optimization problem (supervised regression / classification):

$$f^* = \arg\min_{f \in \mathcal{F}} \ \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)) + \Omega(f)$$

$\mathcal{F}$ is the **class of predictors**

$L : X \times X \to \mathbb{R}$ is the **loss function**

$\Omega : \mathcal{F} \to \mathbb{R}$ is the **regularizer**



$\to$ use regularizer to avoid overfitting

# Empirical Risk Minimization

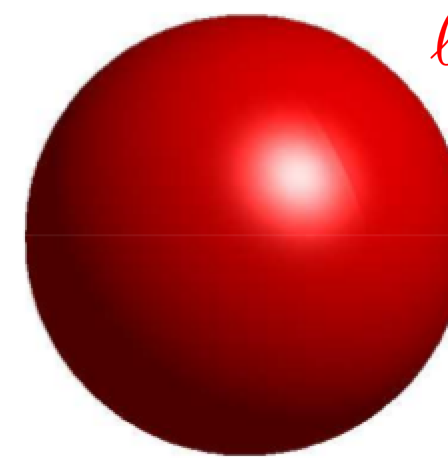
Optimization problem (supervised regression / classification):

$$f^* = \arg\min_{f \in \mathcal{F}} \; \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)) + \Omega(f)$$

$\mathcal{F}$ is the **class of predictors**

$L : X \times X \to \mathbb{R}$ is the **loss function**

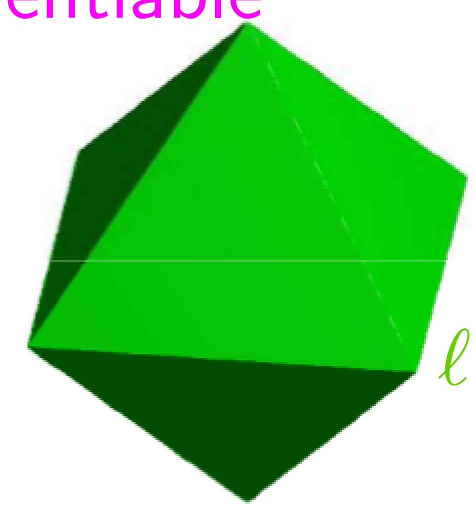
$\Omega : \mathcal{F} \to \mathbb{R}$ is the **regularizer**



$\ell_2$

$$\mathcal{F} = \{f_w : w \in \mathbb{R}^d\}$$

| $\Omega(w)$ | Name |
|---|---|
| $\lVert w \rVert_2^2$ | $\ell_2$ (Tikhonov) $\to$ differentiable |
| $\lVert w \rVert_1$ | $\ell_1$ (LASSO) $\to$ sparse |
| $\alpha \lVert w \rVert_2^2 + (1-\alpha)\lVert w \rVert_1$ | elastic net |



$\ell_1$

# Empirical Risk Minimization

Optimization problem (supervised regression / classification):

$$f^* = \arg\min_{f \in \mathcal{F}} \ \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)) + \Omega(f)$$

$\mathcal{F}$ is the **class of predictors**

$L : X \times X \to \mathbb{R}$ is the **loss function**

$\Omega : \mathcal{F} \to \mathbb{R}$ is the **regularizer**

Complexity of the minimization grows with the one of $\mathcal{F}$

Easy to control when $\mathcal{F}$ is a Reproducing Kernel Hilbert Space

# Reproducing Kernel Hilbert Space

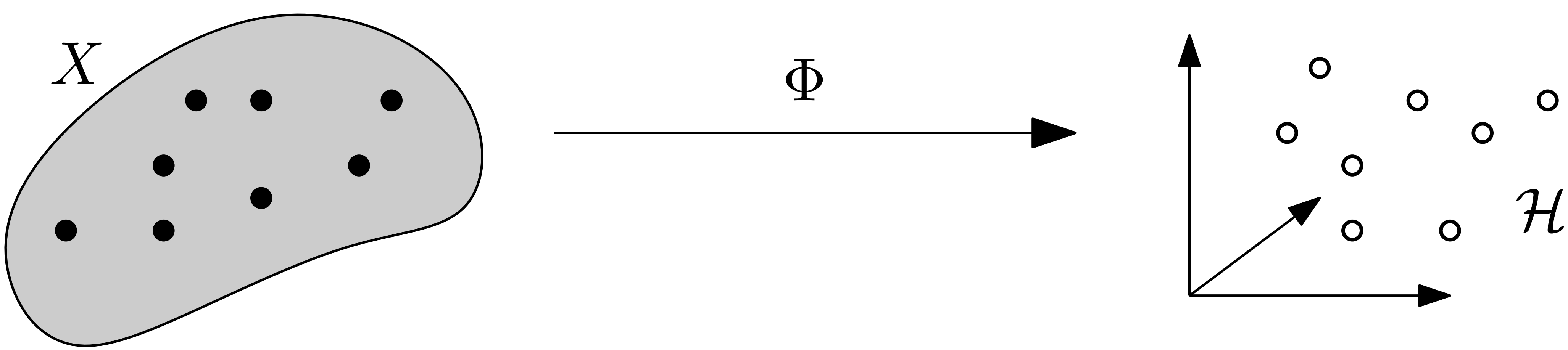
**Def:** Let $\mathcal{H} \subset \mathbb{R}^X$ Hilbert, with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$
Then, $\mathcal{H}$ is a **RKHS** on $X$ if $\exists \Phi : X \to \mathcal{H}$ s.t.:

$$\forall x \in X, \forall f \in \mathcal{H}, \; f(x) = \langle f, \Phi(x) \rangle_{\mathcal{H}}$$

*reproducing* property

Terminology:

- **feature space** $\mathcal{H}$, **feature map** $\Phi$
- **feature vector** $\Phi(x)$
- **kernel** $k = \langle \Phi(\cdot), \Phi(\cdot) \rangle_{\mathcal{H}} : X \times X \to \mathbb{R}$



# Reproducing Kernel Hilbert Space

**Def:** Let $\mathcal{H} \subset \mathbb{R}^X$ Hilbert, with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$
Then, $\mathcal{H}$ is a **RKHS** on $X$ if $\exists \Phi : X \to \mathcal{H}$ s.t.:

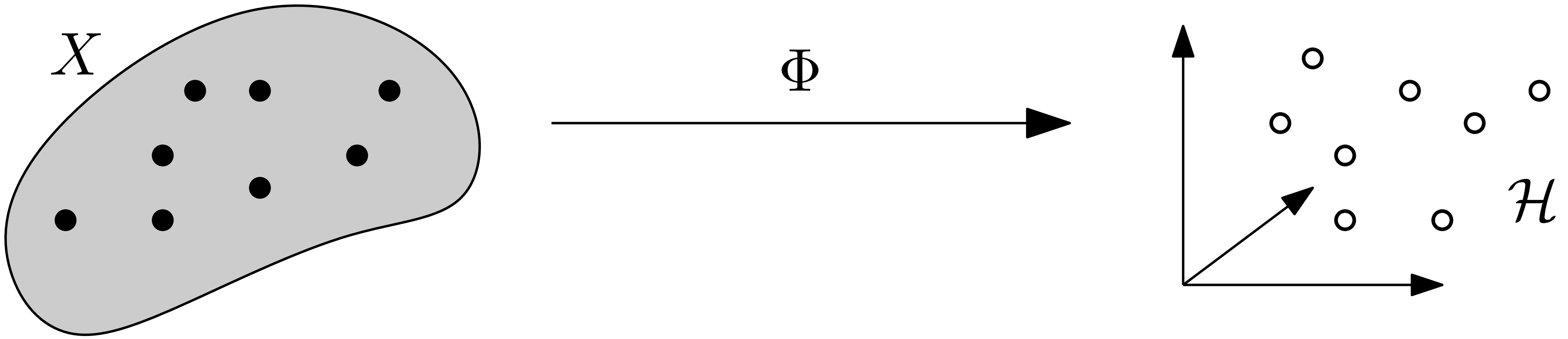
$$\forall x \in X, \forall f \in \mathcal{H}, \; f(x) = \langle f, \Phi(x) \rangle_{\mathcal{H}}$$

*reproducing* property

Terminology:

- **feature space** $\mathcal{H}$, **feature map** $\Phi$
- **feature vector** $\Phi(x)$
- **kernel** $k = \langle \Phi(\cdot), \Phi(\cdot) \rangle_{\mathcal{H}} : X \times X \to \mathbb{R}$

Case $X$ Hilbert space:
$\mathcal{H} = X^*$, $\Phi(x) = \langle x, \cdot \rangle_X$
$\Phi$ isometric isomorphism [Riesz]
$\langle \cdot, \cdot \rangle_{\mathcal{H}} := \langle \Phi^{-1}(\cdot), \Phi^{-1}(\cdot) \rangle_X$



# Reproducing Kernel Hilbert Space

**Def:** Let $\mathcal{H} \subset \mathbb{R}^X$ Hilbert, with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$
Then, $\mathcal{H}$ is a **RKHS** on $X$ if $\exists \Phi : X \to \mathcal{H}$ s.t.:

$$\forall x \in X,\ \forall f \in \mathcal{H},\ f(x) = \langle f, \Phi(x) \rangle_{\mathcal{H}}$$

*reproducing* property

**Prop:** Given $X$, the kernel of a RKHS on $X$ is unique.
Conversely, $k$ is the kernel of at most one RKHS on $X$.

$\rightsquigarrow \Phi(x) = k(x, \cdot)$

# Reproducing Kernel Hilbert Space

**Def:** Let $\mathcal{H} \subset \mathbb{R}^X$ Hilbert, with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$
Then, $\mathcal{H}$ is a **RKHS** on $X$ if $\exists \Phi : X \to \mathcal{H}$ s.t.:

$$\forall x \in X, \forall f \in \mathcal{H}, \; f(x) = \langle f, \Phi(x) \rangle_{\mathcal{H}}$$

*reproducing* property

**Prop:** Given $X$, the kernel of a RKHS on $X$ is unique.
Conversely, $k$ is the kernel of at most one RKHS on $X$.

**Thm:** [Moore 1950] $k : X \times X \to \mathbb{R}$ is a kernel iff it is *positive (semi-)definite*, i.e. $\forall n \in \mathbb{N}$, $\forall x_1, \cdots, x_n \in X$, the Gram matrix $(k(x_i, x_j))_{i,j}$ is positive semi-definite.

# Reproducing Kernel Hilbert Space

**Def:** Let $\mathcal{H} \subset \mathbb{R}^X$ Hilbert, with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$
Then, $\mathcal{H}$ is a **RKHS** on $X$ if $\exists \Phi : X \to \mathcal{H}$ s.t.:

$$\forall x \in X, \forall f \in \mathcal{H}, \; f(x) = \langle f, \Phi(x) \rangle_{\mathcal{H}}$$

*reproducing* property

**Prop:** Given $X$, the kernel of a RKHS on $X$ is unique.
Conversely, $k$ is the kernel of at most one RKHS on $X$.

**Thm:** [Moore 1950] $k : X \times X \to \mathbb{R}$ is a kernel iff it is *positive (semi-)definite*, i.e. $\forall n \in \mathbb{N}, \forall x_1, \cdots, x_n \in X$, the Gram matrix $(k(x_i, x_j))_{i,j}$ is positive semi-definite.

**Examples in** $X = (\mathbb{R}^d, \langle \cdot, \cdot \rangle)$**:**

- linear: $k(x,y) = \langle x, y \rangle$ $\quad \mathcal{H} = (\mathbb{R}^d)^*, \; \Phi(x) = \langle x, \cdot \rangle$
- polynomial: $k(x,y) = (1 + \langle x, y \rangle)^N = \sum_{n_1 + \cdots + n_d = N} \binom{N}{n_1, \cdots, n_d} \underbrace{x_1^{n_1} \cdots x_d^{n_d}}_{\propto \Phi(x)} y_1^{n_1} \cdots y_d^{n_d}$
- Gaussian: $k(x,y) = \exp\left(-\frac{\|x-y\|_2^2}{2\sigma^2}\right)$, $\sigma > 0$. $\quad \mathcal{H} \subset L_2(\mathbb{R}^d)$

# Reproducing Kernel Hilbert Space

**Def:** Let $\mathcal{H} \subset \mathbb{R}^X$ Hilbert, with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$
Then, $\mathcal{H}$ is a **RKHS** on $X$ if $\exists \Phi : X \to \mathcal{H}$ s.t.:

$$\forall x \in X,\ \forall f \in \mathcal{H},\ f(x) = \langle f, \Phi(x) \rangle_{\mathcal{H}}$$

*reproducing* property

**Thm:** (Representer) [Kimeldorf, Wahba 1971] [Schölkopf et al 2001]
Given RKHS $\mathcal{H}$ with kernel $k$, there is a function $f^* \in \mathcal{H}$ minimizing

$$\frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)) + \Omega(\|f\|_{\mathcal{H}})$$

of the form $f^*(\cdot) = \sum_{j=1}^{n} \alpha_j k(x_j, \cdot)$, where $\alpha_1, \cdots, \alpha_n \in \mathbb{R}$.

# Reproducing Kernel Hilbert Space

**Def:** Let $\mathcal{H} \subset \mathbb{R}^X$ Hilbert, with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$
Then, $\mathcal{H}$ is a **RKHS** on $X$ if $\exists \Phi : X \to \mathcal{H}$ s.t.:

$$\forall x \in X,\ \forall f \in \mathcal{H},\ f(x) = \langle f, \Phi(x) \rangle_{\mathcal{H}}$$

*reproducing* property

**Thm:** (Representer) [Kimeldorf, Wahba 1971] [Schölkopf et al 2001]
Given RKHS $\mathcal{H}$ with kernel $k$, there is a function $f^* \in \mathcal{H}$ minimizing

$$\tfrac{1}{n} \textstyle\sum_{i=1}^n L(y_i, f(x_i)) + \Omega(\|f\|_{\mathcal{H}})$$

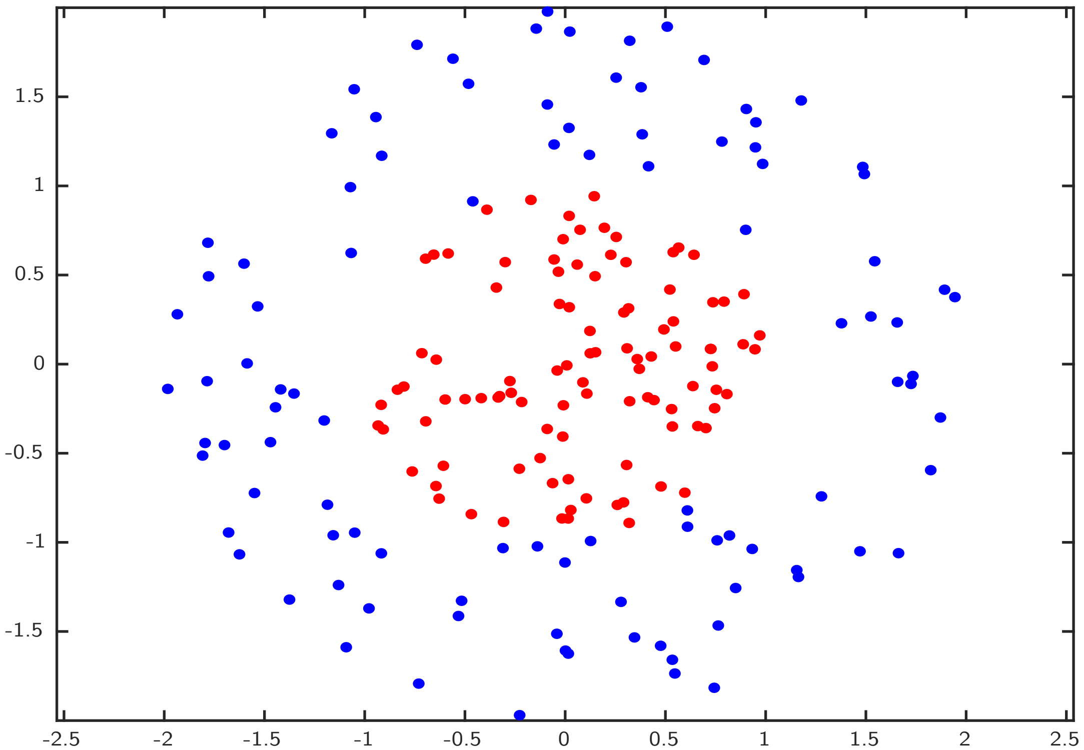
of the form $f^*(\cdot) = \sum_{j=1}^n \alpha_j k(x_j, \cdot)$, where $\alpha_1, \cdots, \alpha_n \in \mathbb{R}$.

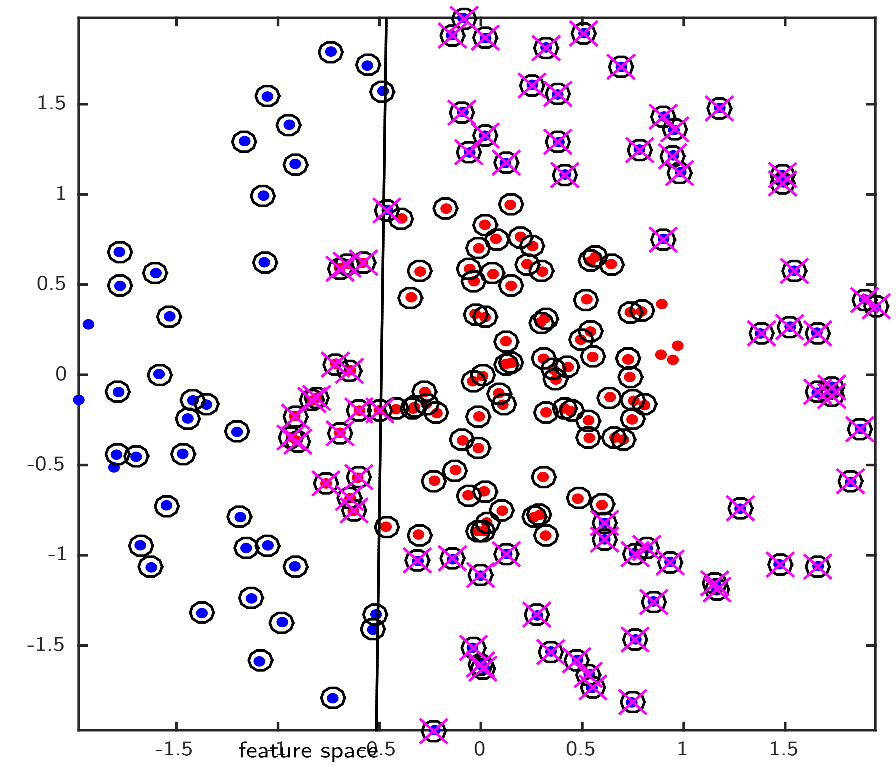
$$\rightsquigarrow \quad \arg\min_{\alpha} \frac{1}{n} \sum_{i=1}^n L\left( y_i, \sum_{j=1}^n \alpha_j k(x_j, x_i) \right) + \Omega\left( \sum_{i,j=1}^n \alpha_i \alpha_j k(x_i, x_j) \right)$$

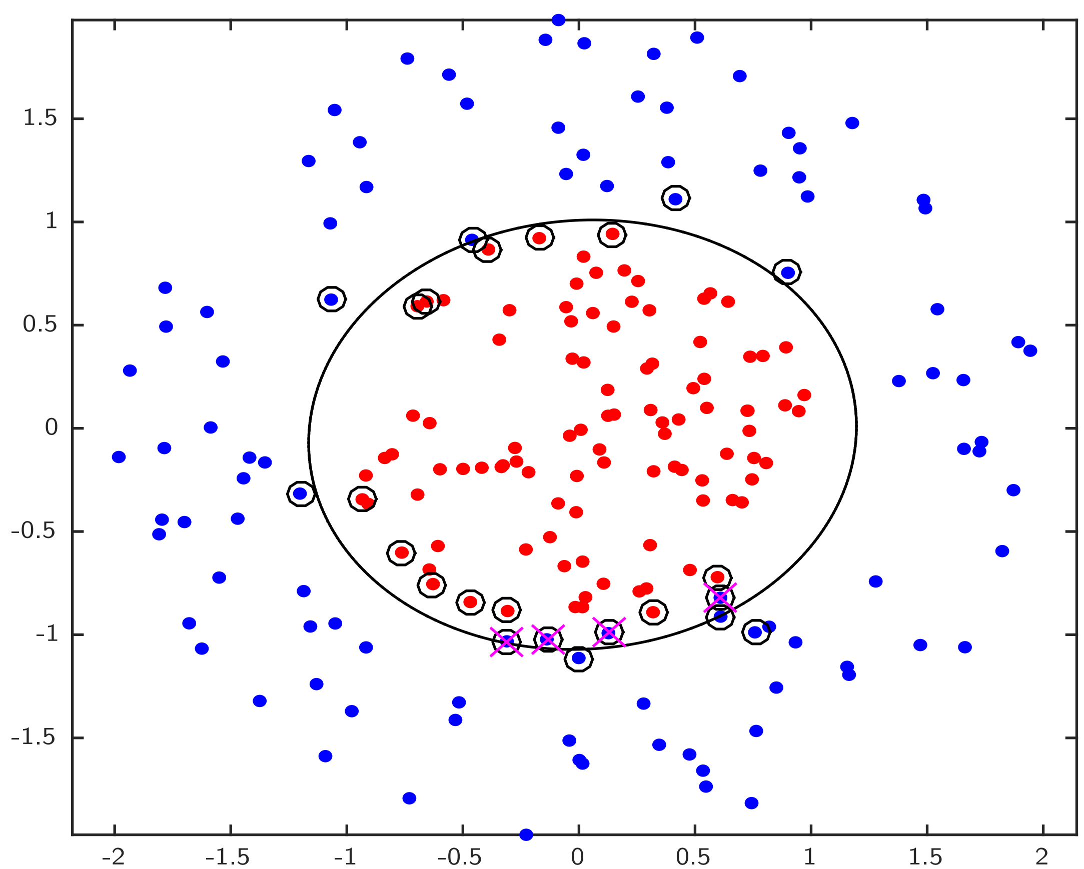
where $\alpha = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix}$ and $K = (k(x_i, x_j))_{ij}$

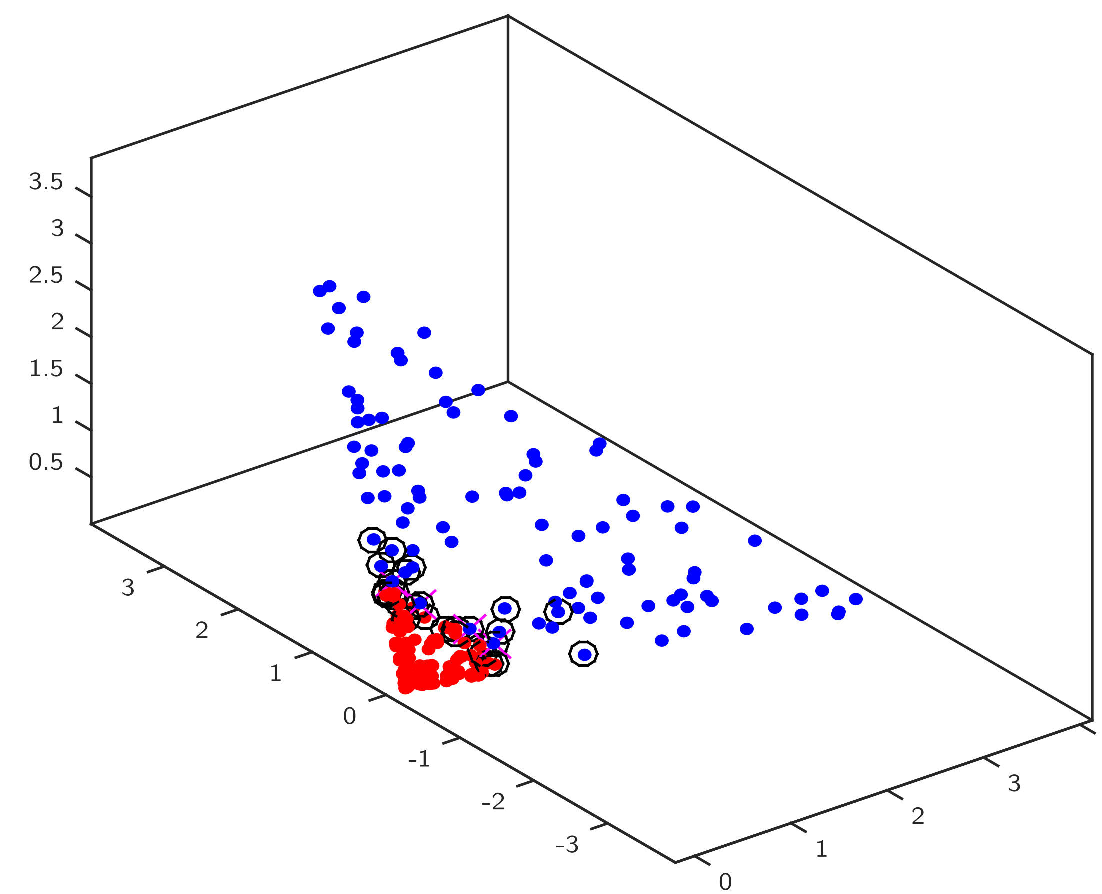
only the $k(x_i, x_j)$ are required to minimize (**kernel trick**)

# Kernel Trick



# Building kernels

Three approaches:

- build kernel from kernels (algebraic operations)

  - **sum of kernels** $\longleftrightarrow$ **concatenation of feature spaces**

$$k_1(x,y) + k_2(x,y) = \left\langle \begin{pmatrix} \Phi_1(x) \\ \Phi_2(x) \end{pmatrix}, \begin{pmatrix} \Phi_1(y) \\ \Phi_2(y) \end{pmatrix} \right\rangle$$

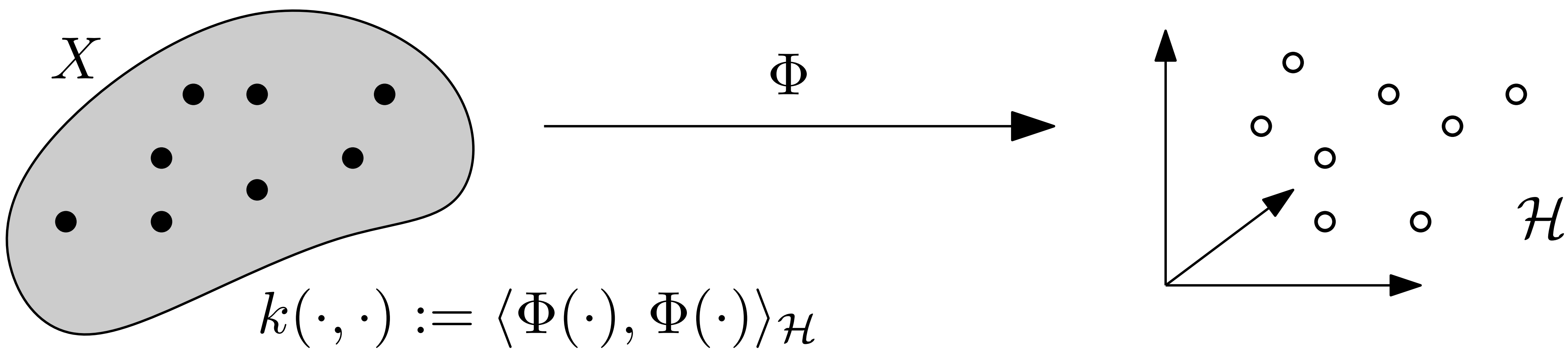
  - **product of kernels** $\longleftrightarrow$ **tensor product of feature spaces**

$$k_1(x,y)k_2(x,y) = \left\langle \Phi_1(x)\Phi_2(x)^T, \Phi_1(y)\Phi_2(y)^T \right\rangle$$

# Building kernels

Three approaches:

- build kernel from kernels (algebraic operations)
- define explicit feature map $\Phi : X \to \mathcal{H}$ (vectorization)



# Building kernels

Three approaches:

- build kernel from kernels (algebraic operations)
- define explicit feature map $\Phi : X \to \mathcal{H}$ (vectorization)
- define kernel from metric via radial basis function

**Thm:** [Kimeldorf, Wahba 1971]

If $d : X \times X \to \mathbb{R}_+$ symmetric is *conditionally negative semidefinite*, i.e.:

$$\forall n \in \mathbb{N},\ \forall x_1, \cdots, x_n \in X,\ \sum_{i=1}^{n} \alpha_i = 0 \Longrightarrow \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j \, d(x_i, x_j) \leq 0,$$

then $k(x, y) = \exp\left(-\frac{d(x,y)}{2\sigma^2}\right)$ is positive definite for all $\sigma > 0$.

# Building kernels

Three approaches:

- build kernel from kernels (algebraic operations)
- define explicit feature map $\Phi : X \to \mathcal{H}$ (vectorization)
- define kernel from metric via radial basis function

**Thm:** [Kimeldorf, Wahba 1971]

If $d : X \times X \to \mathbb{R}_+$ symmetric is *conditionally negative semidefinite*, i.e.:

$$\forall n \in \mathbb{N},\ \forall x_1, \cdots, x_n \in X,\ \sum_{i=1}^{n} \alpha_i = 0 \Longrightarrow \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j\, d(x_i, x_j) \leq 0,$$

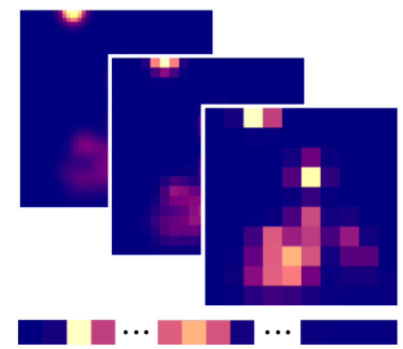
then $k(x, y) = \exp\left(-\frac{d(x,y)}{2\sigma^2}\right)$ is positive definite for all $\sigma > 0$.

**Q:** does this apply to persistence diagrams?

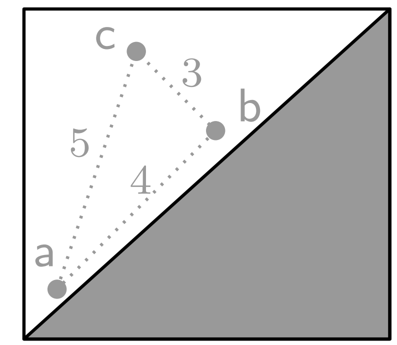
**A:** no, $\mathrm{d}_p$ is **not** cnsd

# Vectorizations for persistence diagrams

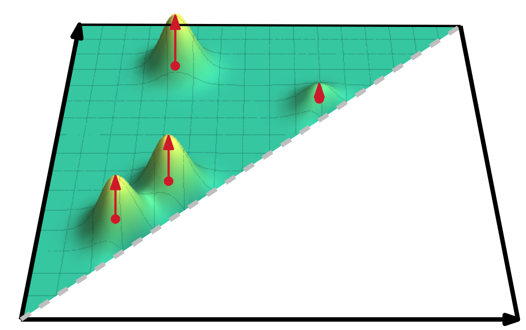
- images [Adams et al. '15]



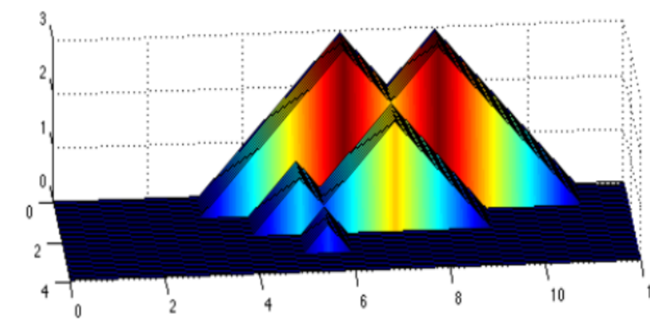
$$\begin{array}{c|ccc} & a & b & c \\ \hline a & 0 & 4 & 5 \\ b & 4 & 0 & 3 \\ c & 5 & 3 & 0 \end{array}$$




- finite metric spaces [Carrière et al. '15]



- landscapes [Bubenik '12] [Bubenik, Dłotko '15]



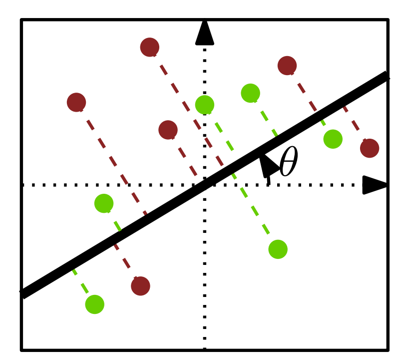
- discrete measures:

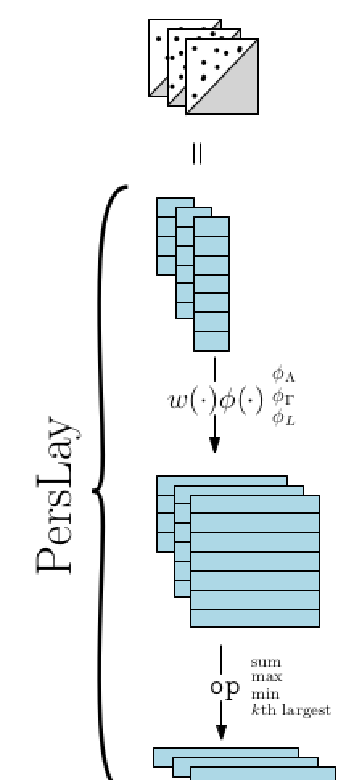
  → histograms [Bendich et al. '14]

  → convolutions [Chepushtanova et al. '15] [Kusano et al. '16-'17]

  → heat diffusion [Reininghaus et al. '15] [Kwit et al. '15]

  → sliced Wasserstein distances [Carrière et al. '17]




- test functions

  → polynomials [Di Fabio, Ferri '15] [Kališnik '16]

  → deep sets [Carrière et al. '20]

# Theoretical guarantees

| | images | metric spaces | polynomials | landscapes | discrete measures |
|---|---|---|---|---|---|
| ambient Hilbert space | $(\mathbb{R}^d, \lVert . \rVert_2)$ | $(\mathbb{R}^d, \lVert . \rVert_2)$ | $\ell_2(\mathbb{R})$ | $L_2(\mathbb{N} \times \mathbb{R})$ | $L_2(\mathbb{R}^2)$ |
| positive (semi-)definiteness | ✓ | ✓ | ✓ | ✓ | ✓ |
| $\lVert \phi(\cdot) - \phi(\cdot) \rVert_{\mathcal{H}} \leq C(\mathrm{d}_p)$ | ✓ | ✓ | ✓ | ✓ | ✓ |
| $\lVert \phi(\cdot) - \phi(\cdot) \rVert_{\mathcal{H}} \geq c(\mathrm{d}_p)$ | ✗ | ✗ | ✗ | ✗ | ✗ |
| injectivity | ✗ | ✗ | ✓ | ✓ | ✓ |
| universality | ✗ | ✗ | ✗ | ✗ | ✓ |
| algorithmic cost | f. map: $O(n^2)$<br>kernel: $O(d)$ | f. map: $O(n^2)$<br>kernel: $O(d)$ | f. map: $O(nd)$<br>kernel: $O(d)$ | $O(n^2)$ | $O(n^2)$ |

# Theoretical guarantees

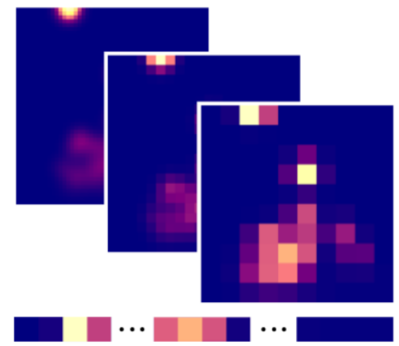
| | images | metric spaces | polynomials | landscapes | discrete measures |
|---|---|---|---|---|---|
| ambient Hilbert space | $(\mathbb{R}^d, \|.\|_2)$ | $(\mathbb{R}^d, \|.\|_2)$ | $\ell_2(\mathbb{R})$ | $L_2(\mathbb{N} \times \mathbb{R})$ | $L_2(\mathbb{R}^2)$ |
| positive (semi-)definiteness | ✓ | ✓ | ✓ | ✓ | ✓ |
| $\|\phi(\cdot) - \phi(\cdot)\|_{\mathcal{H}} \leq C(\mathrm{d}_p)$ | ✓ | ✓ | ✓ | ✓ | ✓ |
| $\|\phi(\cdot) - \phi(\cdot)\|_{\mathcal{H}} \geq c(\mathrm{d}_p)$ | ✗ | ✗ | ✗ | ✗ | ✗ |
| injectivity | ✗ | ✗ | ✓ | ✓ | ✓ |
| universality | ✗ | ✗ | ✗ | ✗ | ✓ |
| algorithmic cost | f. map: $O(n^2)$<br>kernel: $O(d)$ | f. map: $O(n^2)$<br>kernel: $O(d)$ | f. map: $O(nd)$<br>kernel: $O(d)$ | $O(n^2)$ | $O(n^2)$ |

# Theoretical guarantees

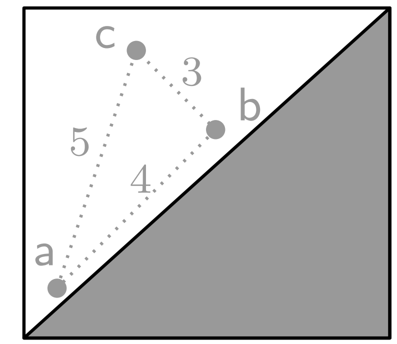
| | images | metric spaces | polynomials | landscapes | discrete measures |
|---|---|---|---|---|---|
| ambient Hilbert space | $(\mathbb{R}^d, \|.\|_2)$ | $(\mathbb{R}^d, \|.\|_2)$ | $\ell_2(\mathbb{R})$ | $L_2(\mathbb{N} \times \mathbb{R})$ | $L_2(\mathbb{R}^2)$ |
| positive (semi-)definiteness | ✓ | ✓ | ✓ | ✓ | ✓ |
| $\|\phi(\cdot) - \phi(\cdot)\|_{\mathcal{H}} \leq C(\mathrm{d}_p)$ | ✓ | ✓ | ✓ | ✓ | ✓ |
| $\boxed{\|\phi(\cdot) - \phi(\cdot)\|_{\mathcal{H}} \geq c(\mathrm{d}_p)}$ | ✗ | ✗ | ✗ | ✗ | ✗ |
| injectivity | ✗ | ✗ | ✓ | ✓ | ✓ |
| universality | ✗ | ✗ | ✗ | ✗ | ✓ |
| algorithmic cost | f. map: $O(n^2)$<br>kernel: $O(d)$ | f. map: $O(n^2)$<br>kernel: $O(d)$ | f. map: $O(nd)$<br>kernel: $O(d)$ | $O(n^2)$ | $O(n^2)$ |

# Vectorizations for persistence diagrams

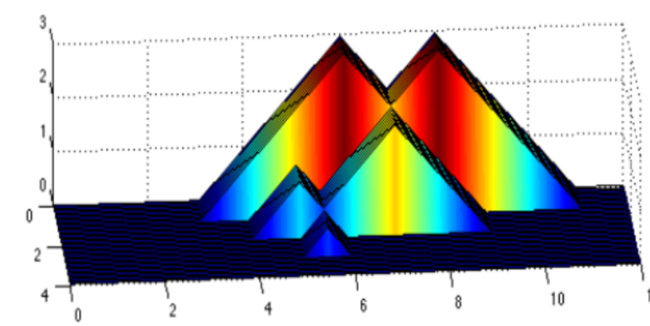
- images [Adams et al. '15]



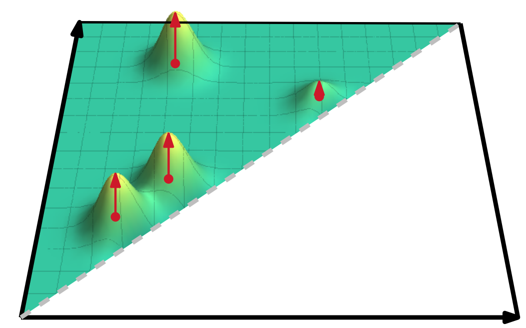
$$\begin{array}{c|ccc} & a & b & c \\ \hline a & 0 & 4 & 5 \\ b & 4 & 0 & 3 \\ c & 5 & 3 & 0 \end{array}$$



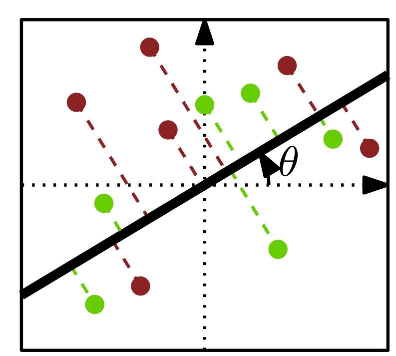

- finite metric spaces [Carrière et al. '15]

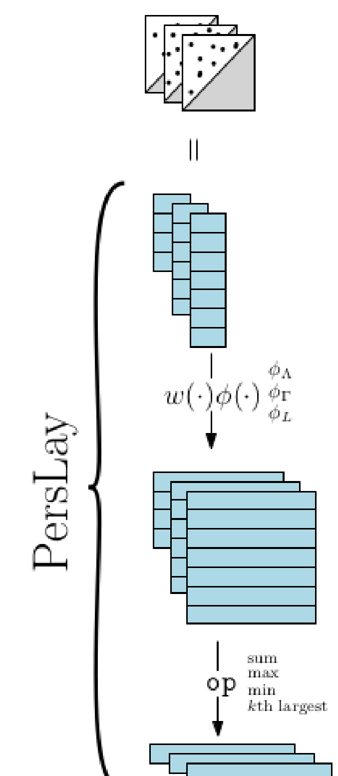


- landscapes [Bubenik '12] [Bubenik, Dłotko '15]



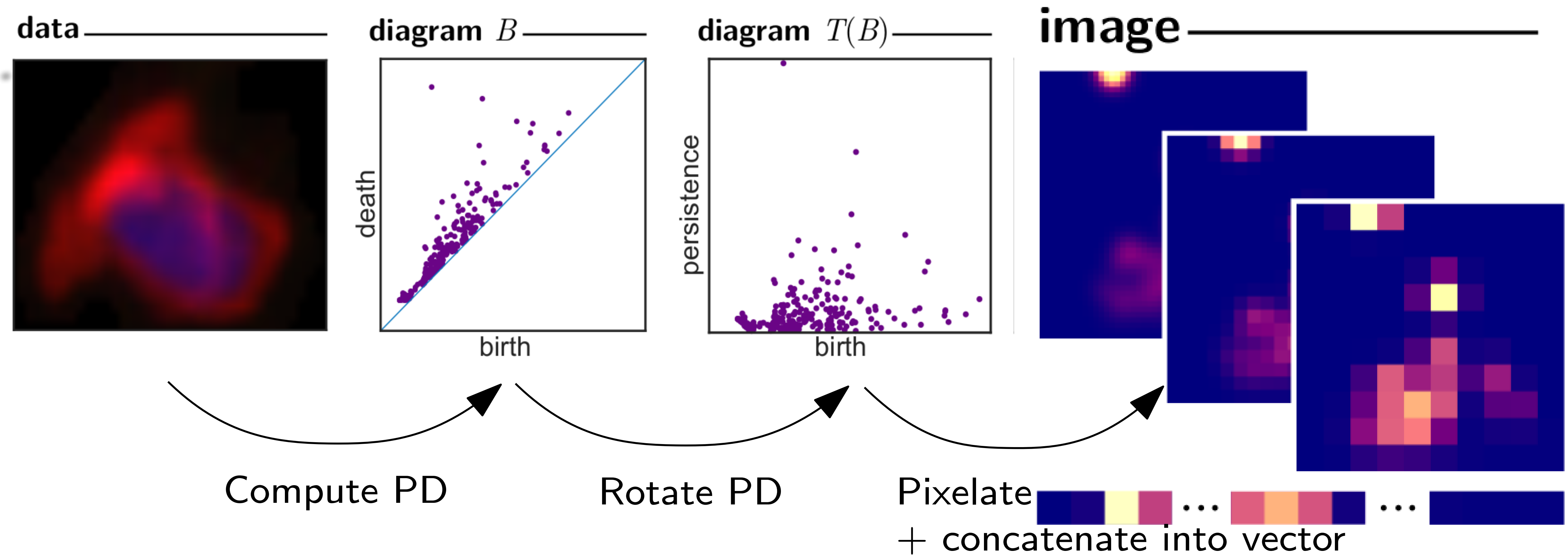
- discrete measures:
  → histograms [Bendich et al. '14]
  → convolutions [Chepushtanova et al. '15] [Kusano et al. '16-'17]
  → heat diffusion [Reininghaus et al. '15] [Kwit et al. '15]
  → sliced Wasserstein distances [Carrière et al. '17]



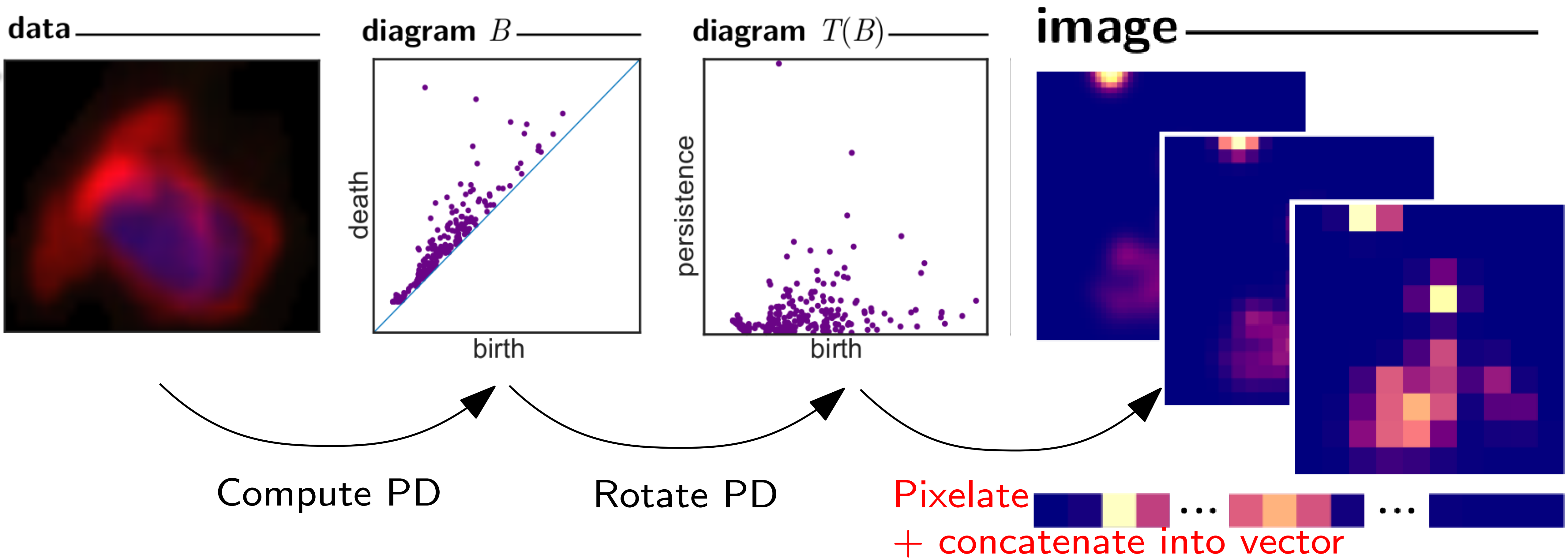



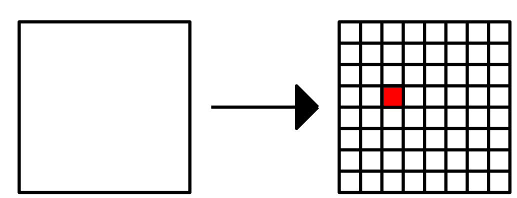

- test functions
  → polynomials [Di Fabio, Ferri '15] [Kališnik '16]
  → deep sets [Carrière et al. '20]

# Persistence Images [Adams et al. 2017]



# Persistence Images [Adams et al. 2017]

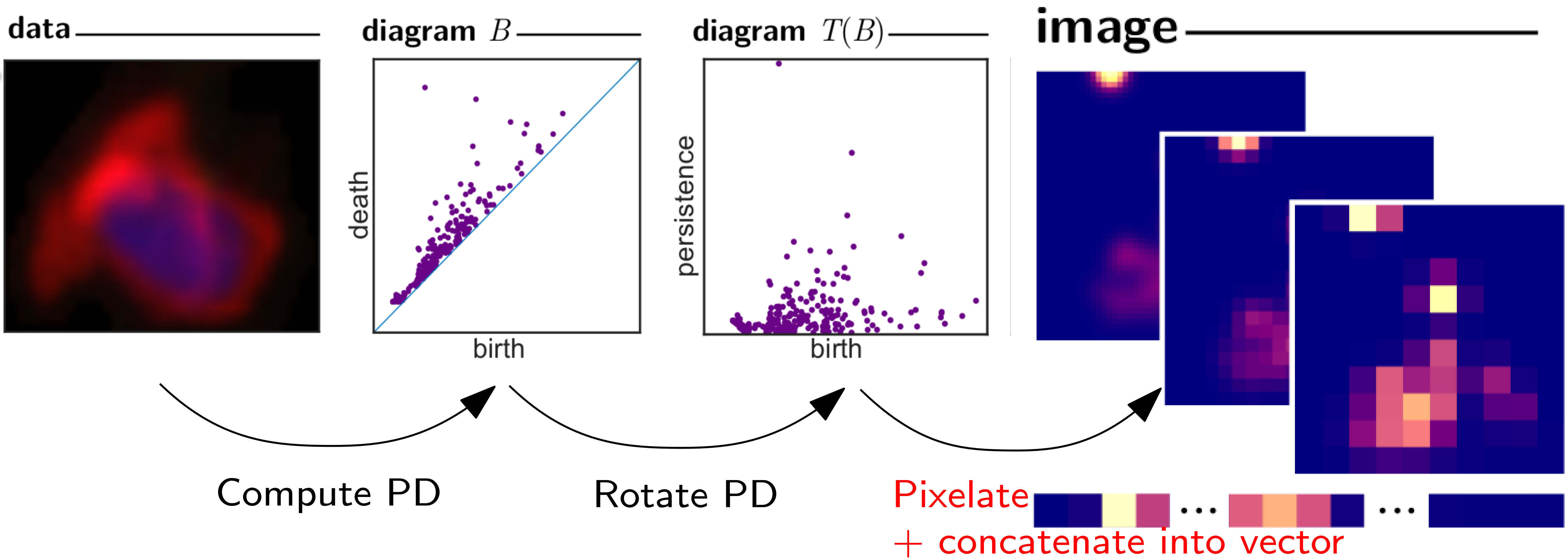



Discretize plane into one or several grid(s): 

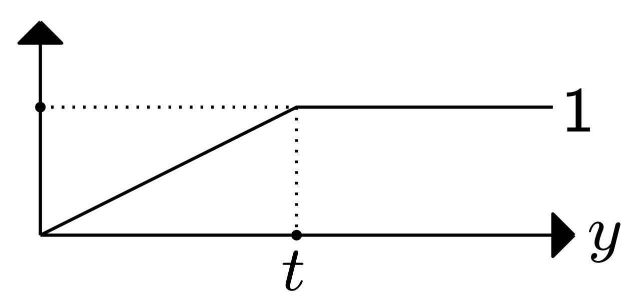
For each pixel $P$, compute $I(P) = \#\ \mathrm{Dgm} \cap P$

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

# Persistence Images [Adams et al. 2017]



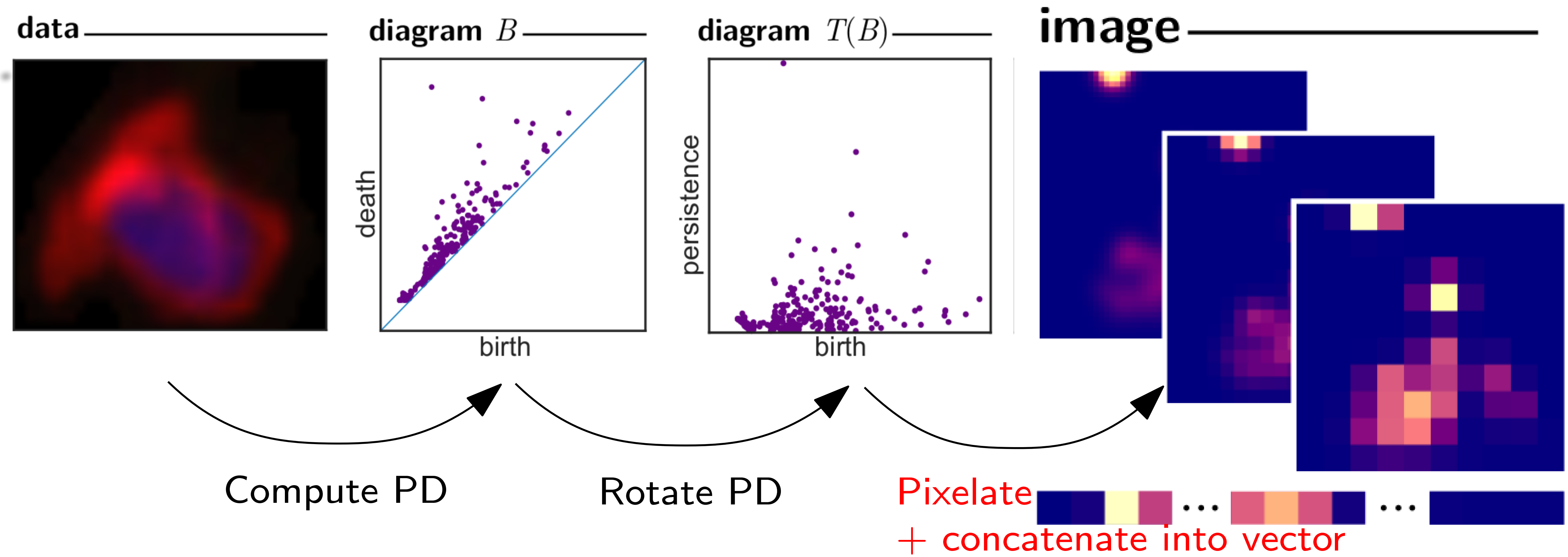

Stability → weigh points: $w_t(x,y) =$




→ blur image

(convolve with Gaussian)

# Persistence Images [Adams et al. 2017]

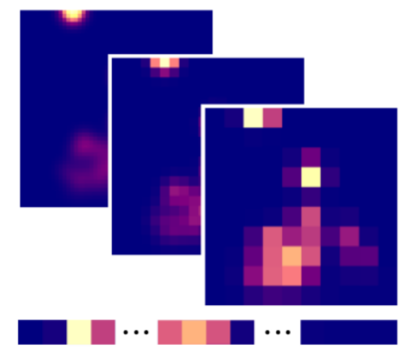


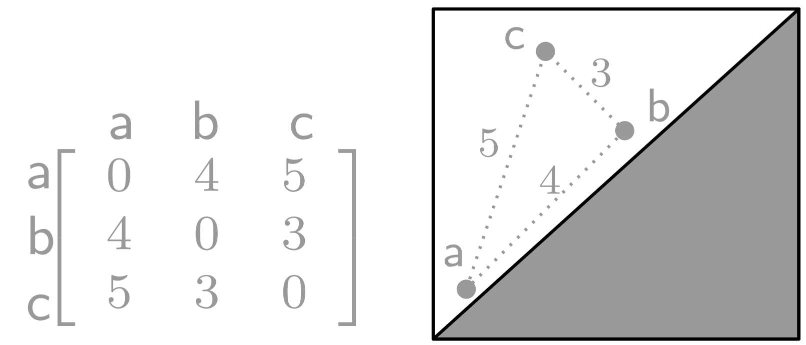

**Prop:** [Adams et al. 2017]

- $\|\mathrm{PI}(\mathrm{Dgm}) - \mathrm{PI}(\mathrm{Dgm}')\|_\infty \leq C(w, \phi_p)\, \mathrm{d}_1(\mathrm{Dgm}, \mathrm{Dgm}')$
- $\|\mathrm{PI}(\mathrm{Dgm}) - \mathrm{PI}(\mathrm{Dgm}')\|_2 \leq \sqrt{d} C(w, \phi_p)\, \mathrm{d}_1(\mathrm{Dgm}, \mathrm{Dgm}')$

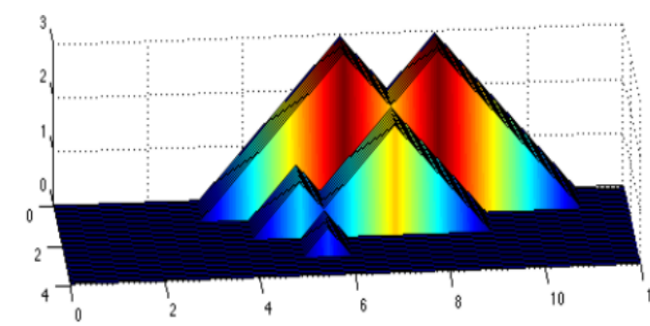
# Vectorizations for persistence diagrams

- images [Adams et al. '15]

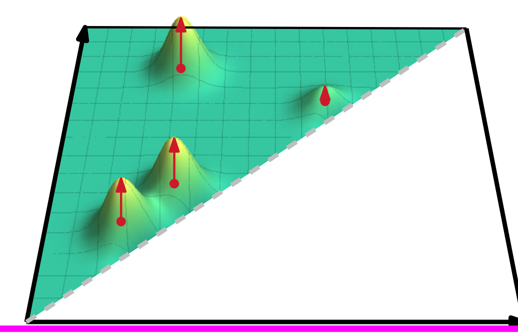





- finite metric spaces [Carrière et al. '15]



- landscapes [Bubenik '12] [Bubenik, Dłotko '15]
- discrete measures:

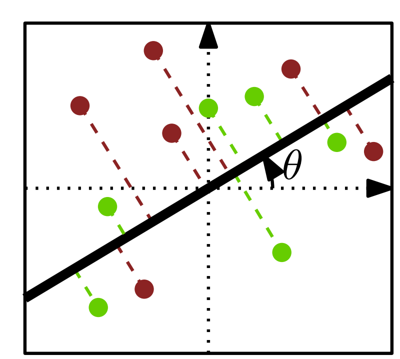


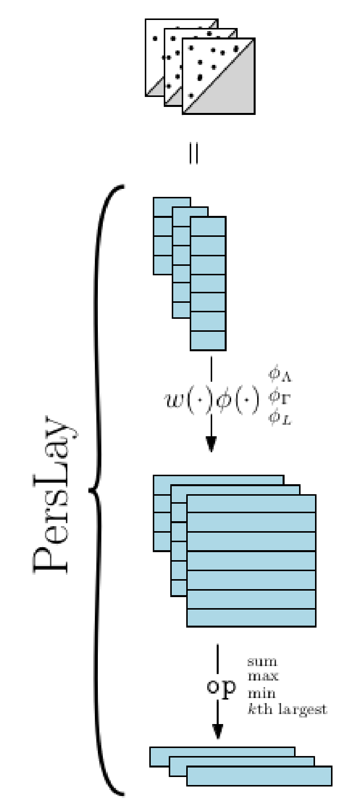
  → histograms [Bendich et al. '14]

  → convolutions [Chepushtanova et al. '15] [Kusano et al. '16-'17]

  → heat diffusion [Reininghaus et al. '15] [Kwit et al. '15]

  → sliced Wasserstein distances [Carrière et al. '17]



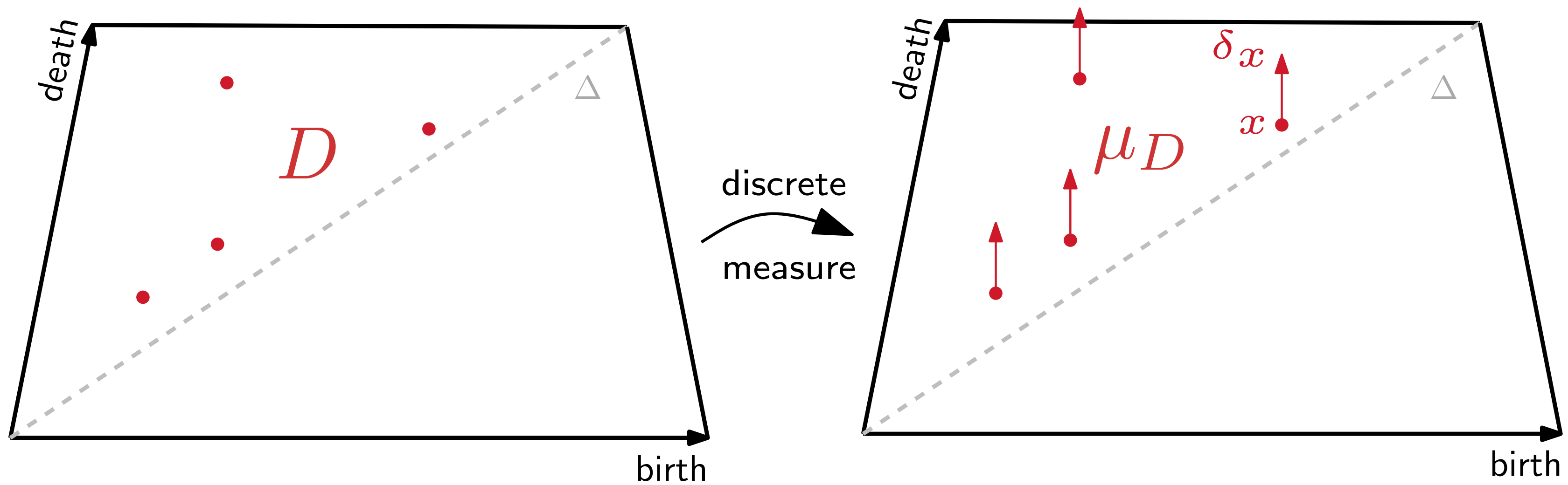

- test functions

  → polynomials [Di Fabio, Ferri '15] [Kališnik '16]

  → deep sets [Carrière et al. '20]

# Convolution-based vectorization

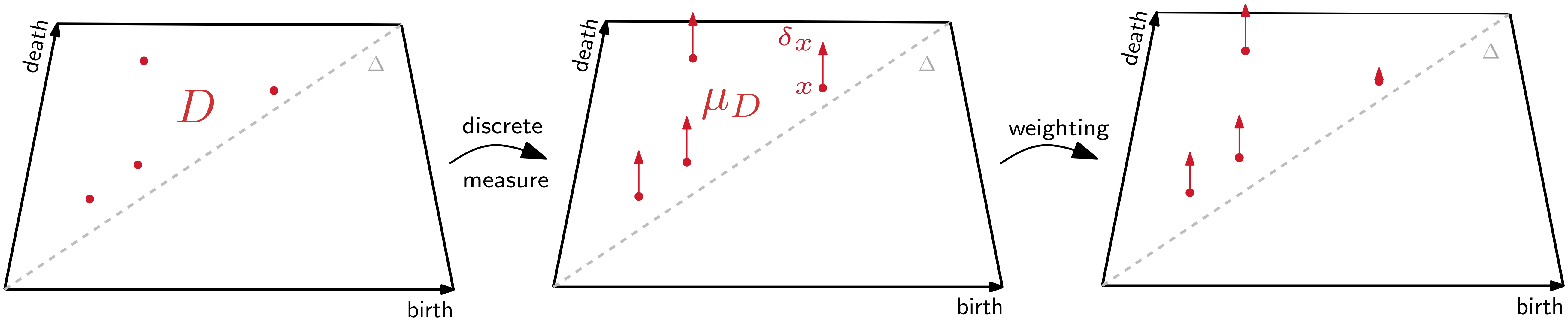
Persistence diagrams as discrete measures:




$$\mu_D := \sum_{x \in D} \delta_x$$

# Convolution-based vectorization

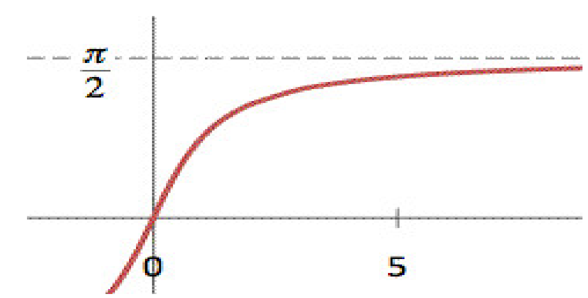
Persistence diagrams as discrete measures:




$\mu_D := \sum_{x\in D} \delta_x$ $\mu_D^w := \sum_{x\in D} w(x)\delta_{\bar{x}}$

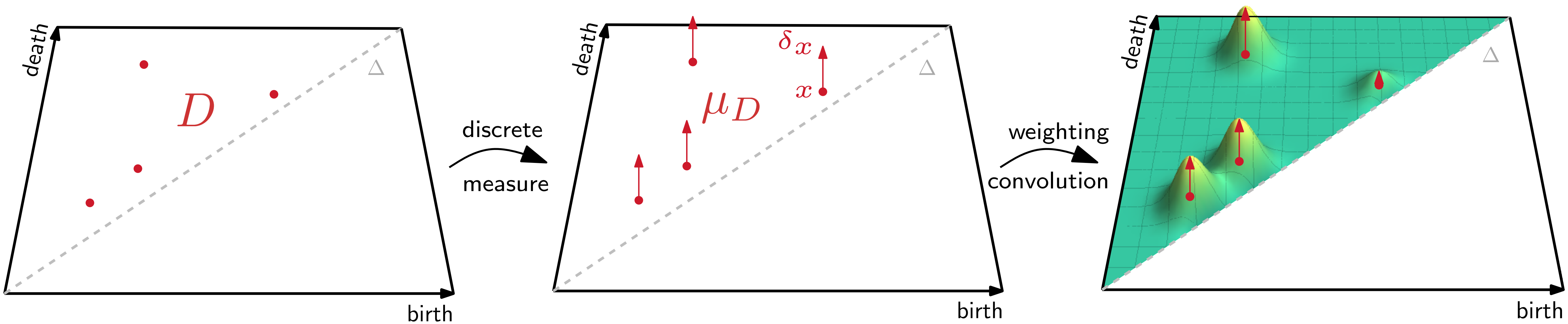
**Pb:** $\mu_D$ is unstable (points on diagonal disappear)

$w(x) := \arctan\left(c\, \mathrm{d}(x,\Delta)^r\right),\ c, r > 0$



# Convolution-based vectorization

Persistence diagrams as discrete measures:




$\mu_D := \sum_{x \in D} \delta_x$

$\mu_D^w := \sum_{x \in D} w(x) \delta_{\bar{x}}$

$\tilde{\mu}_D^w := \mu_D^w * \mathcal{N}(0, \sigma)$

**Pb:** $\mu_D$ is unstable (points on diagonal disappear)

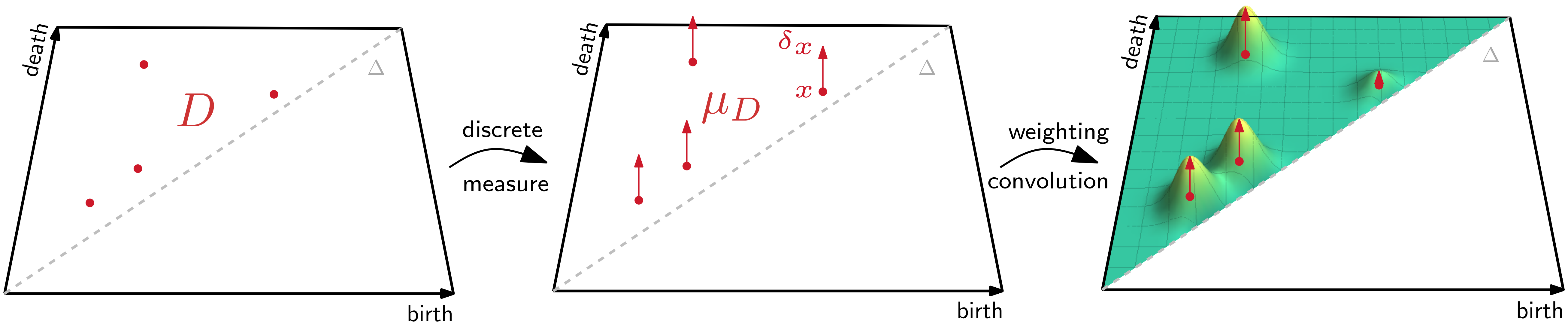
$w(x) := \arctan\left(c\,\mathrm{d}(x, \Delta)^r\right),\ c, r > 0$

**Def:** $\phi(D)$ is the density function of $\mu_D^w * \mathcal{N}(0, \sigma)$ w.r.t. Lebesgue measure:

$$\left(\begin{array}{l} \phi(D) := \dfrac{1}{\sqrt{2\pi}\sigma} \displaystyle\sum_{x \in D} \arctan(c\,\mathrm{d}(x, \Delta)^r) \exp\left(-\dfrac{\|\cdot - x\|^2}{2\sigma^2}\right) \\ k(D, D') := \langle \phi(D),\ \phi(D') \rangle_{L_2(\Delta \times \mathbb{R}_+)} \end{array}\right.$$

# Convolution-based vectorization

Persistence diagrams as discrete measures:




$\mu_D := \sum_{x \in D} \delta_x$

$\mu_D^w := \sum_{x \in D} w(x) \delta_{\bar{x}}$

$\tilde{\mu}_D^w := \mu_D^w * \mathcal{N}(0, \sigma)$

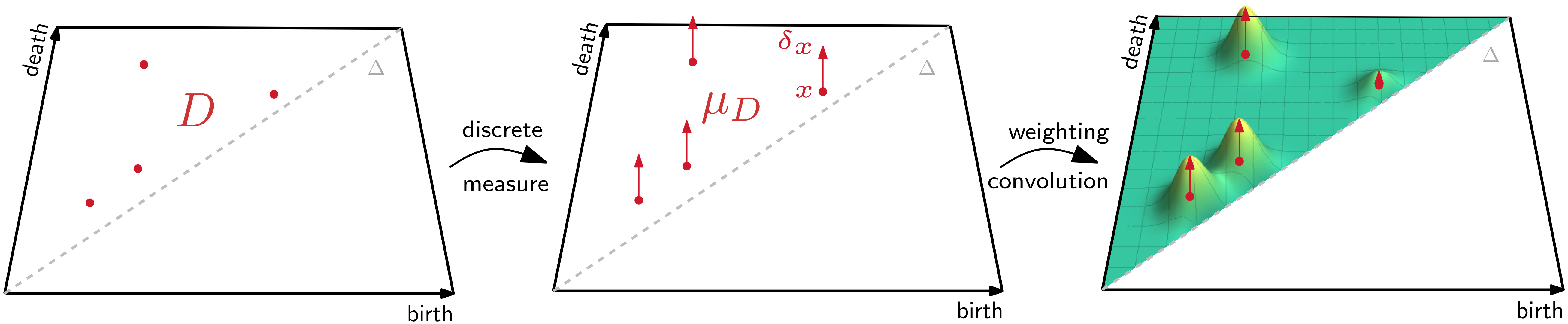
**Prop.:** [Kusano, Fukumisu, Hiraoka 2016-17]

- $\|\phi(D) - \phi(D')\|_{\mathcal{H}} \leq \mathrm{cst}\, d_p(D, D')$.
- $\phi$ is injective and $\exp(k)$ is universal

$$\begin{cases} \phi(D) := \dfrac{1}{\sqrt{2\pi}\sigma} \displaystyle\sum_{x \in D} \arctan(c\, \mathrm{d}(x, \Delta)^r) \exp\left( -\dfrac{\| \cdot - x\|^2}{2\sigma^2} \right) \\ k(D, D') := \langle \phi(D),\, \phi(D') \rangle_{L_2(\Delta \times \mathbb{R}_+)} \end{cases}$$

# Convolution-based vectorization

Persistence diagrams as discrete measures:




$\mu_D := \sum_{x \in D} \delta_x$

$\mu_D^w := \sum_{x \in D} w(x) \delta_{\bar{x}}$

$\tilde{\mu}_D^w := \mu_D^w * \mathcal{N}(0, \sigma)$

**Prop.:** [Kusano, Fukumisu, Hiraoka 2016-17]

- $\|\phi(D) - \phi(D')\|_{\mathcal{H}} \leq \mathrm{cst}\, d_p(D, D')$.
- $\phi$ is injective and $\exp(k)$ is universal

**Pb:** convolution reduces discriminativity $\rightarrow$ use discrete measure instead

$$\left\{\begin{array}{l} \phi(D) := \dfrac{1}{\sqrt{2\pi}\sigma} \displaystyle\sum_{x \in D} \arctan(c\, \mathrm{d}(x, \Delta)^r) \exp\left(-\dfrac{\|\cdot - x\|^2}{2\sigma^2}\right) \\ k(D, D') := \langle \phi(D), \phi(D') \rangle_{L_2(\Delta \times \mathbb{R}_+)} \end{array}\right.$$

# Theoretical guarantees

| | images | metric spaces | polynomials | landscapes | discrete measures |
|---|---|---|---|---|---|
| ambient Hilbert space | $(\mathbb{R}^d, \|.\|_2)$ | $(\mathbb{R}^d, \|.\|_2)$ | $\ell_2(\mathbb{R})$ | $L_2(\mathbb{N} \times \mathbb{R})$ | $L_2(\mathbb{R}^2)$ |
| positive (semi-)definiteness | ✓ | ✓ | ✓ | ✓ | ✓ |
| $\|\phi(\cdot) - \phi(\cdot)\|_{\mathcal{H}} \leq C(\mathrm{d}_p)$ | ✓ | ✓ | ✓ | ✓ | ✓ |
| $\|\phi(\cdot) - \phi(\cdot)\|_{\mathcal{H}} \geq c(\mathrm{d}_p)$ | ✗ | ✗ | ✗ | ✗ | ✗ |
| injectivity | ✗ | ✗ | ✓ | ✓ | ✓ |
| universality | ✗ | ✗ | ✗ | ✗ | ✓ |
| algorithmic cost | f. map: $O(n^2)$<br>kernel: $O(d)$ | f. map: $O(n^2)$<br>kernel: $O(d)$ | f. map: $O(nd)$<br>kernel: $O(d)$ | $O(n^2)$ | $O(n^2)$ |

# One kernel to rule them all...

Sliced Wasserstein Kernel [Carrière, Cuturi, O. 2017]

No feature map

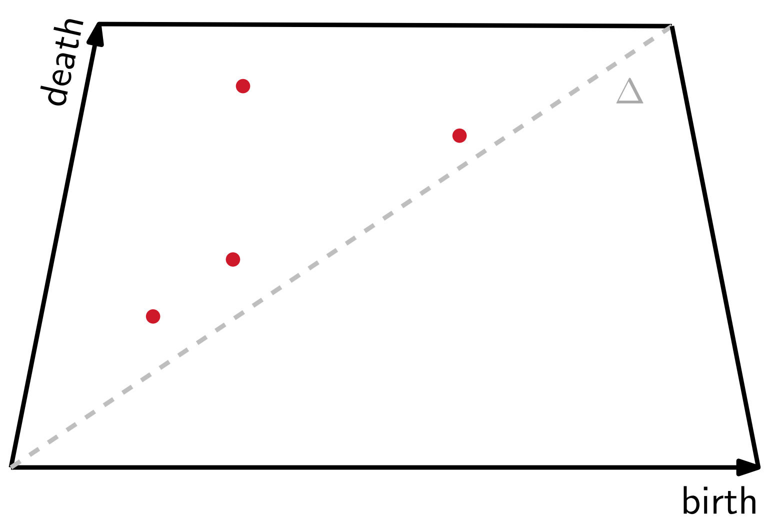
Provably stable

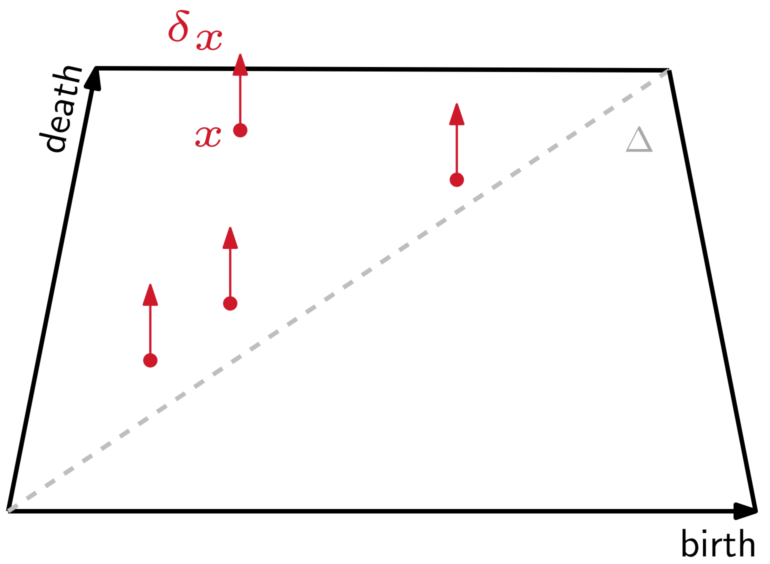
Provably discriminative

Mimicks the Gaussian kernel

View diagrams as discrete measures w/o density functions

# Persistence diagrams as discrete measures (II)

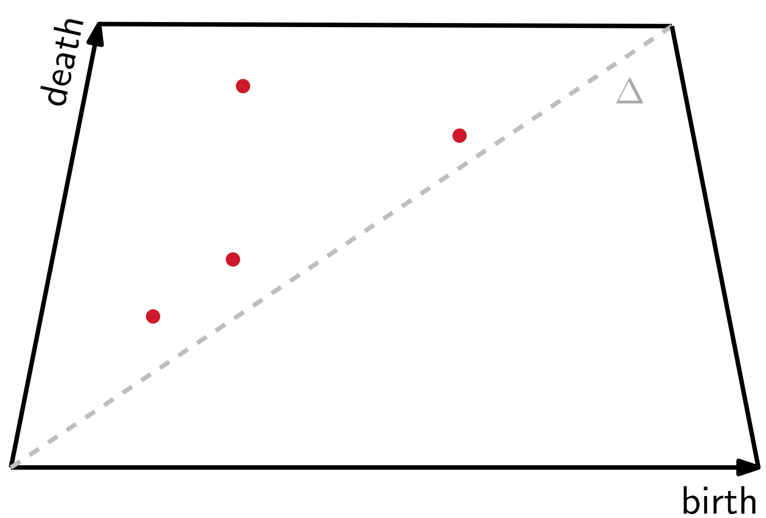


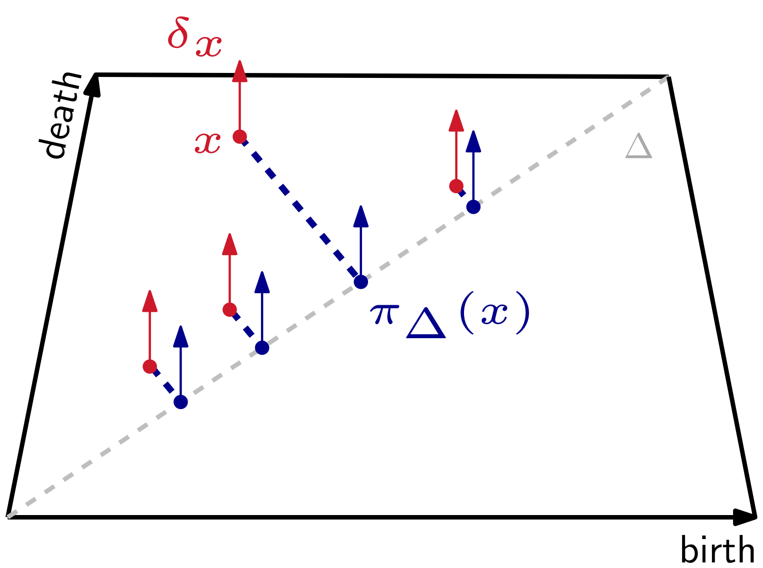




$\mu_D := \sum_{x \in D} \delta_x$

**Pb:** $\mathrm{d}_p(D, D') \not\propto W_p(\mu_D, \mu_{D'})$ ($W_p$ does not even make sense here)

# Persistence diagrams as discrete measures (II)




$\mu_D := \sum_{x \in D} \delta_x$

**Pb:** $\mathrm{d}_p(D, D') \not\propto W_p(\mu_D, \mu_{D'})$ ($W_p$ does not even make sense here)

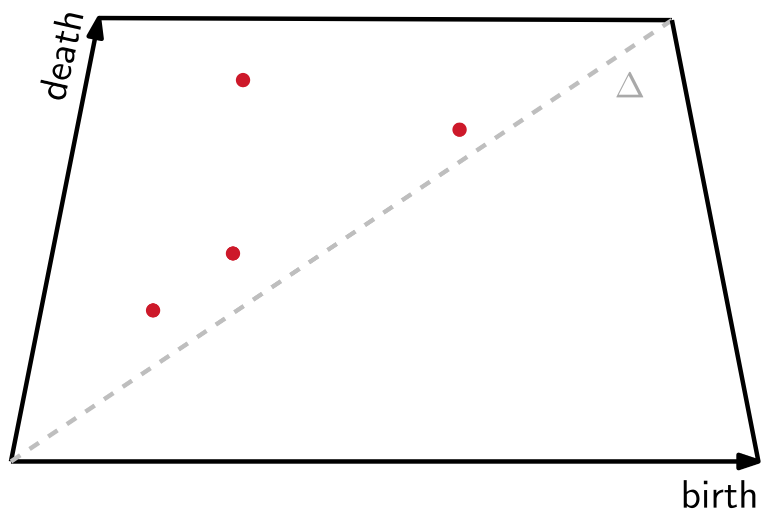
$\rightarrow$ given $D, D'$, let

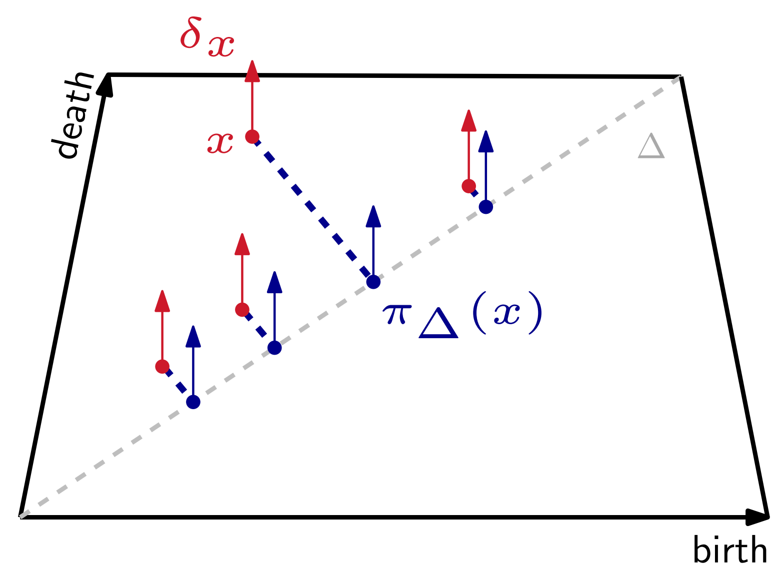
$$\bar{\mu}_D := \sum_{x \in D} \delta_x + \sum_{y \in D'} \delta_{\pi_\Delta(y)}$$

$$\bar{\mu}_{D'} := \sum_{y \in D'} \delta_y + \sum_{x \in D} \delta_{\pi_\Delta(x)}$$

Then, $\mathrm{d}_p(D, D') \leq W_p(\bar{\mu}_D, \bar{\mu}_{D'}) \leq 2\,\mathrm{d}_p(D, D')$

# Persistence diagrams as discrete measures (II)




$\mu_D := \sum_{x \in D} \delta_x$

**Pb:** $\mathrm{d}_p(D, D') \not\propto W_p(\mu_D, \mu_{D'})$ ($W_p$ does not even make sense here)

$\rightarrow$ given $D, D'$, let

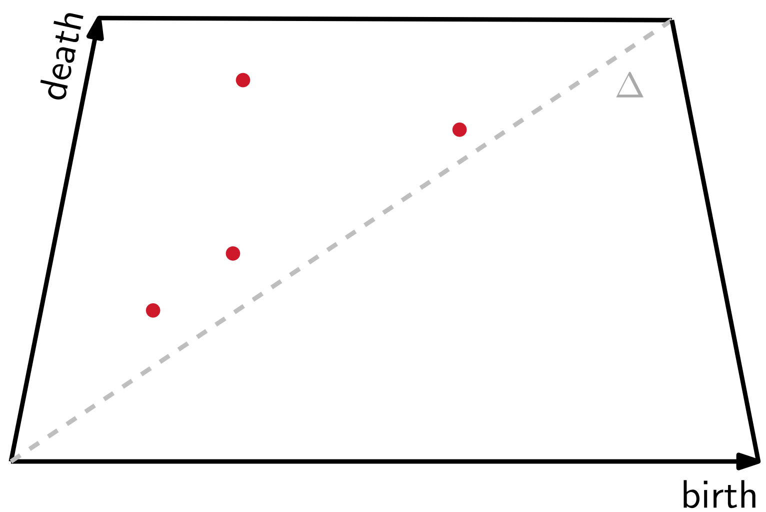
$$\bar{\mu}_D := \sum_{x \in D} \delta_x + \sum_{y \in D'} \delta_{\pi_\Delta(y)}$$

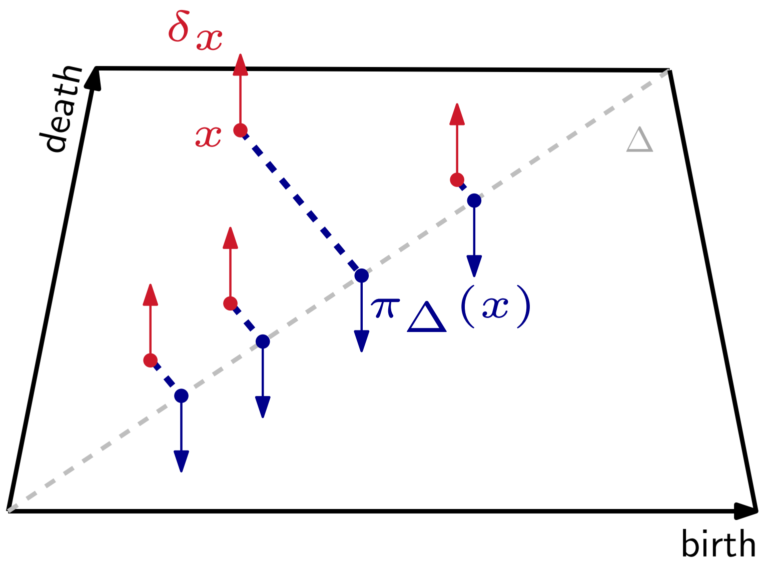
$$\bar{\mu}_{D'} := \sum_{y \in D'} \delta_y + \sum_{x \in D} \delta_{\pi_\Delta(x)}$$

Then, $\mathrm{d}_p(D, D') \leq W_p(\bar{\mu}_D, \bar{\mu}_{D'}) \leq 2\,\mathrm{d}_p(D, D')$

**Pb:** $\bar{\mu}_D$ depends on $D'$

# Persistence diagrams as discrete measures (II)




$\mu_D := \sum_{x \in D} \delta_x$

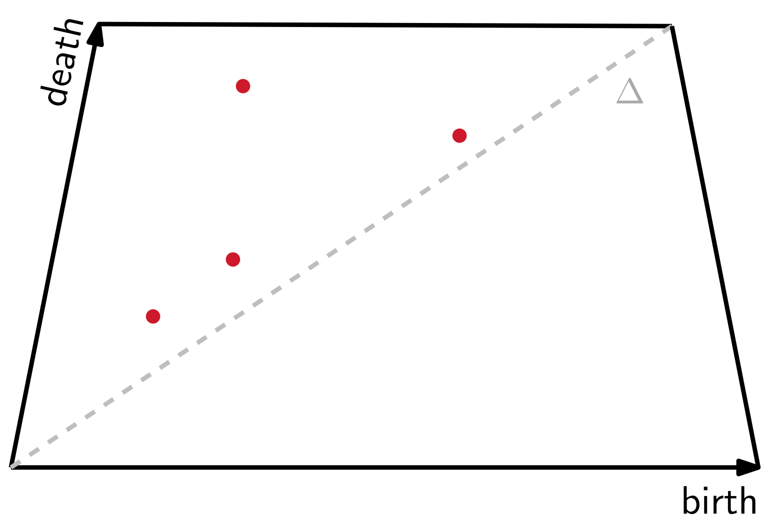
**Pb:** $\mathrm{d}_p(D, D') \not\propto W_p(\mu_D, \mu_{D'})$ ($W_p$ does not even make sense here)

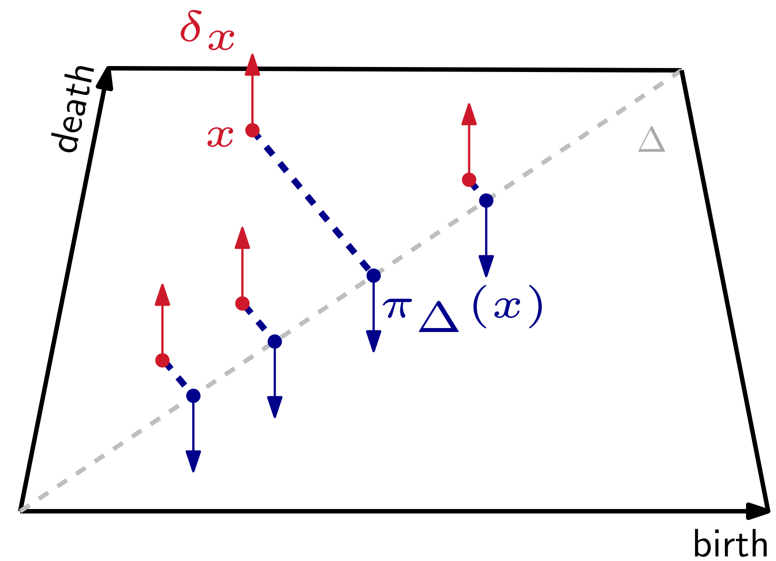
Solution: transfer mass negatively to $\mu_D$:

$$\tilde{\mu}_D := \sum_{x \in D} \delta_x - \sum_{x \in D} \delta_{\pi_\Delta(x)} \quad \in \mathcal{M}_0(\mathbb{R}^2)$$

$\rightarrow$ signed discrete measure of total mass zero

# Persistence diagrams as discrete measures (II)




$\mu_D := \sum_{x \in D} \delta_x$

**Pb:** $\mathrm{d}_p(D, D') \not\propto W_p(\mu_D, \mu_{D'})$ ($W_p$ does not even make sense here)

Solution: transfer mass negatively to $\mu_D$:

$$\tilde{\mu}_D := \sum_{x \in D} \delta_x - \sum_{x \in D} \delta_{\pi_\Delta(x)} \quad \in \mathcal{M}_0(\mathbb{R}^2)$$

$\rightarrow$ signed discrete measure of total mass zero

**metric:** Kantorovich norm $\|\cdot\|_K$

# Persistence diagrams as discrete measures (II)

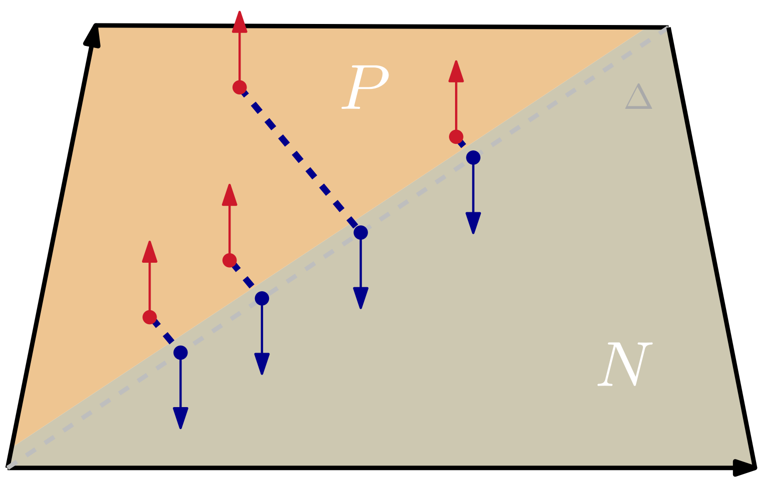
**Hahn decomposition thm.:** For any $\mu \in \mathcal{M}_0(X, \Sigma)$ there exist measurable sets $P, N$ such that:

(i) $P \cup N = X$ and $P \cap N = \emptyset$

(ii) $\mu(B) \geq 0$ for every measureable set $B \subseteq P$

(iii) $\mu(B) \leq 0$ for every measureable set $B \subseteq N$

Moreover, the decomposition is essentially unique.




$\forall B \in \Sigma$, let $\mu^+(B) := \mu(B \cap P)$ and $\mu^-(B) := -\mu(B \cap N) \in \mathcal{M}_+(X)$

**Def.:** $\|\mu\|_K := \mathbf{W_1}(\mu^+, \mu^-)$

**Prop.:** $\forall \mu, \nu \in \mathcal{M}_0(X)$, $\quad W_1(\mu^+ + \nu^-, \nu^+ + \mu^-) = \|\mu - \nu\|_K$

# Persistence diagrams as discrete measures (II)

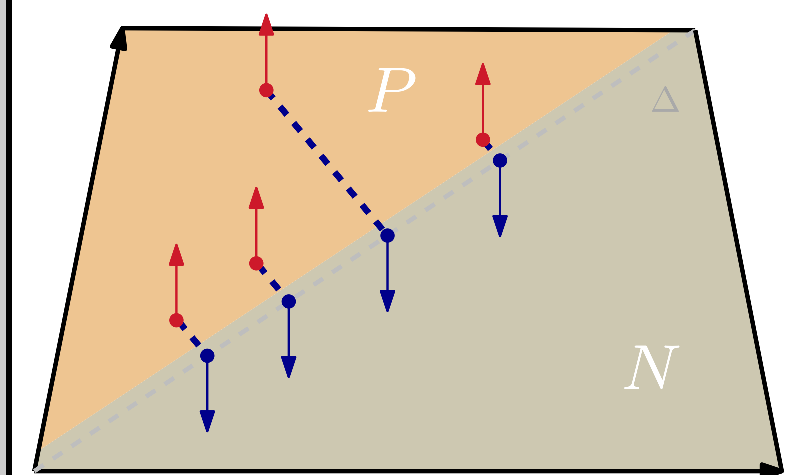
**Hahn decomposition thm.:** For any $\mu \in \mathcal{M}_0(X, \Sigma)$ there exist measurable sets $P, N$ such that:

(i) $P \cup N = X$ and $P \cap N = \emptyset$

(ii) $\mu(B) \geq 0$ for every measureable set $B \subseteq P$

(iii) $\mu(B) \leq 0$ for every measureable set $B \subseteq N$

Moreover, the decomposition is essentially unique.




$\forall B \in \Sigma$, let $\mu^+(B) := \mu(B \cap P)$ and $\mu^-(B) := -\mu(B \cap N) \in \mathcal{M}_+(X)$

**Def.:** $\|\mu\|_K := \mathbf{W_1}(\mu^+, \mu^-)$

**Prop.:** $\forall \mu, \nu \in \mathcal{M}_0(X)$, $\quad W_1(\underbrace{\mu^+ + \nu^-}_{\bar{\mu}_D}, \underbrace{\nu^+ + \mu^-}_{\bar{\mu}_{D'}}) = \|\underset{\substack{\uparrow \\ \tilde{\mu}_D}}{\mu} - \underset{\substack{\uparrow \\ \tilde{\mu}_{D'}}}{\nu}\|_K$

for persistence diagrams:

$$W_1(\bar{\mu}_D, \bar{\mu}_{D'}) = \|\tilde{\mu}_D - \tilde{\mu}_{D'}\|_K$$

# A Wasserstein Gaussian kernel for PDs?

**Thm.:** [Kimeldorf, Wahba 1971]
If $d : X \times X \to \mathbb{R}_+$ symmetric is *conditionally negative semidefinite*, i.e.:

$$\forall n \in \mathbb{N},\ \forall x_1, \cdots, x_n \in X,\ \sum_{i=1}^{n} \alpha_i = 0 \Longrightarrow \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j\, d(x_i, x_j) \leq 0,$$

then $k(x, y) := \exp\left(-\frac{d(x,y)}{2\sigma^2}\right)$ is positive semidefinite.

**Pb:** $W_1$ is not cnsd, neither is $d_1$

Solutions:

- relax the measures (e.g. convolution)
- relax the metric (e.g. regularization, slicing)

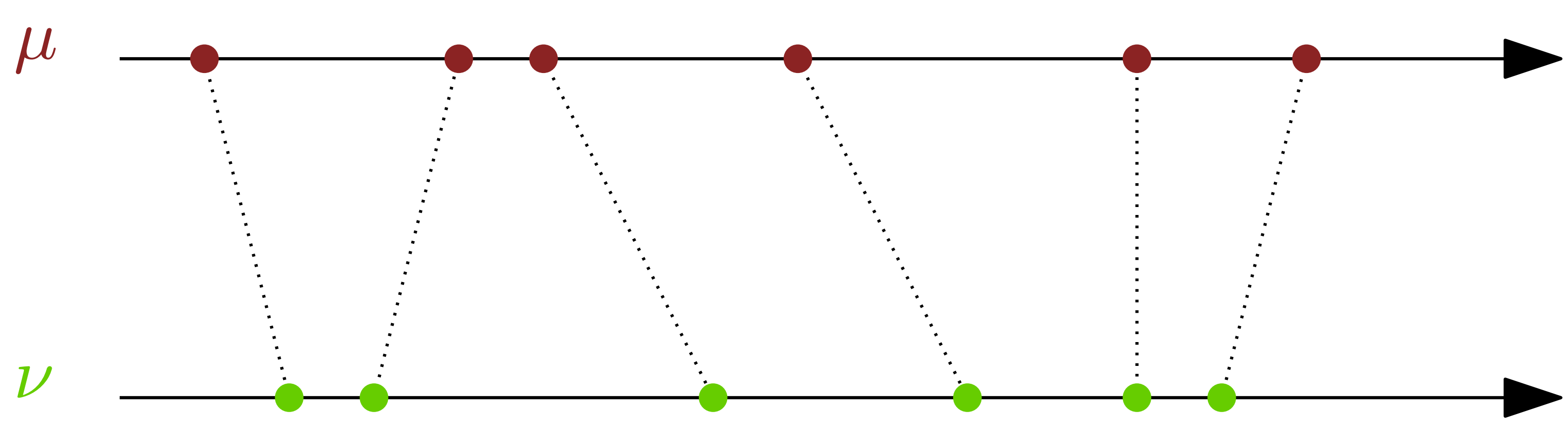
# Sliced Wasserstein metric

**Special case:** $X = \mathbb{R}$, $\mu, \nu$ discrete measures of mass $n$

$\mu := \sum_{i=1}^{n} \delta_{x_i}, \;\; \nu := \sum_{i=1}^{n} \delta_{y_i}$

Sort the atoms of $\mu, \nu$ along the real line: $x_i \leq x_{i+1}$ and $y_i \leq y_{i+1}$ for all $i$

Then: $W_1(\mu, \nu) = \sum_{i=1}^{n} |x_i - y_i| = \|(x_1, \cdots, x_n) - (y_1, \cdots, y_n)\|_1$



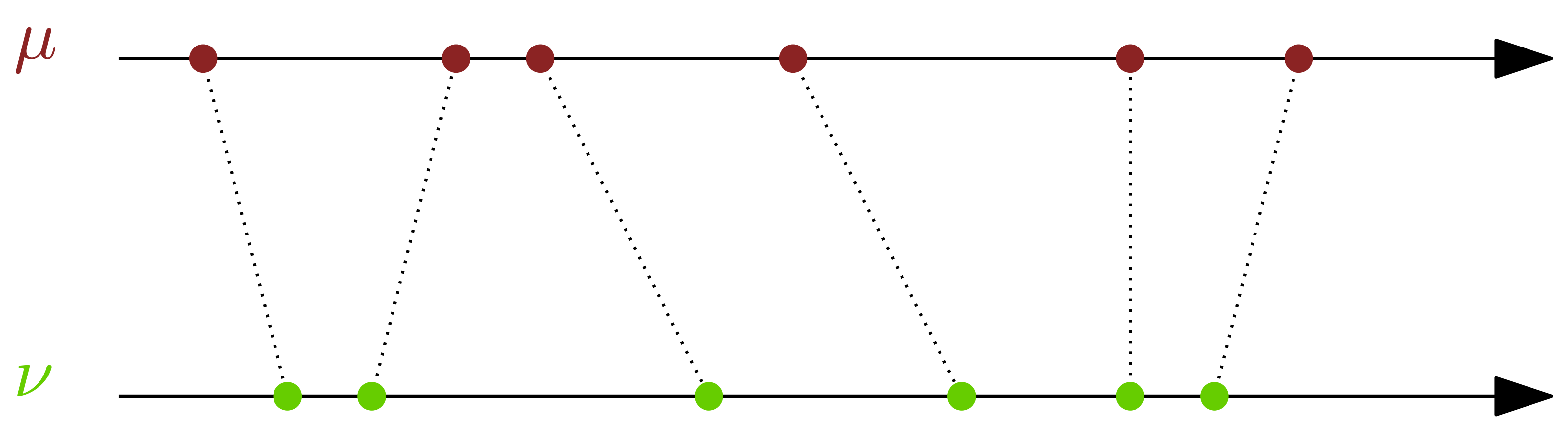
# Sliced Wasserstein metric

**Special case:** $X = \mathbb{R}$, $\mu, \nu$ discrete measures of mass $n$

$\mu := \sum_{i=1}^n \delta_{x_i}$, $\nu := \sum_{i=1}^n \delta_{y_i}$

Sort the atoms of $\mu, \nu$ along the real line: $x_i \leq x_{i+1}$ and $y_i \leq y_{i+1}$ for all $i$

Then: $W_1(\mu, \nu) = \sum_{i=1}^n |x_i - y_i| = \|(x_1, \cdots, x_n) - (y_1, \cdots, y_n)\|_1$



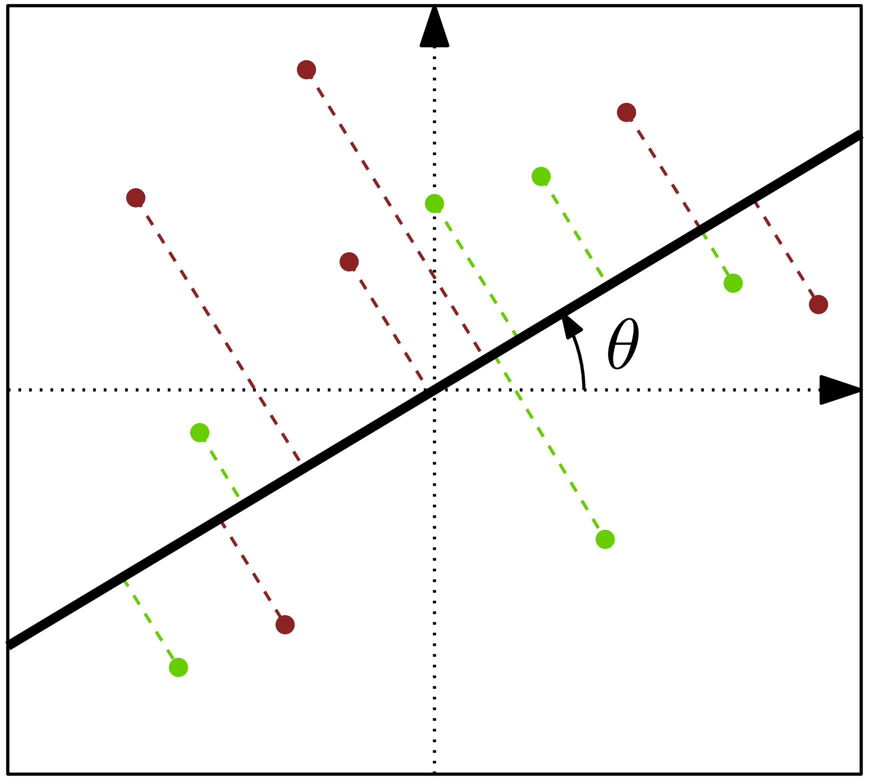

$\rightarrow$ $W_1$ is cnsd and easy to compute (same with $\|\cdot\|_K$ for signed measures)

# Sliced Wasserstein metric

**Def (sliced Wasserstein distance):** for $\mu, \nu \in \mathcal{M}_+(\mathbb{R}^2)$,

$$SW_1(\mu,\nu) := \frac{1}{2\pi} \int_{\theta\in\mathbb{S}^1} W_1(\pi_\theta\#\mu,\ \pi_\theta\#\nu)\, d\theta$$

where $\pi_\theta$ = orthogonal projection onto line passing through origin with angle $\theta$.




$\rightarrow$ from integral geometry: $\int_{\mathrm{Gr}(1,2)} \cdots$

# Sliced Wasserstein metric

**Def (sliced Wasserstein distance):** for $\mu, \nu \in \mathcal{M}_+(\mathbb{R}^2)$,

$$SW_1(\mu,\nu) := \frac{1}{2\pi} \int_{\theta \in \mathbb{S}^1} W_1(\pi_\theta \# \mu,\, \pi_\theta \# \nu)\, d\theta$$

where $\pi_\theta =$ orthogonal projection onto line passing through origin with angle $\theta$.

**Props:** (inherited from $W_1$ over $\mathbb{R}$) [Rabin, Peyré, Delon, Bernot 2011]

- satisfies the axioms of a metric

- well-defined barycenters, fast to compute via stochastic gradient descent, etc.

- conditionally negative semidefinite

# Sliced Wasserstein kernel

**Def:** Given $\sigma > 0$, for any $\mu, \nu \in \mathcal{M}_+(\mathbb{R}^2)$:

$$k_{SW}(\mu, \nu) := \exp\left(-\frac{SW_1(\mu, \nu)}{2\sigma^2}\right)$$

**Corollary:** [Kolouri, Zou, Rohde](from $SW$ cnsd)
$k_{SW}$ is positive semidefinite.

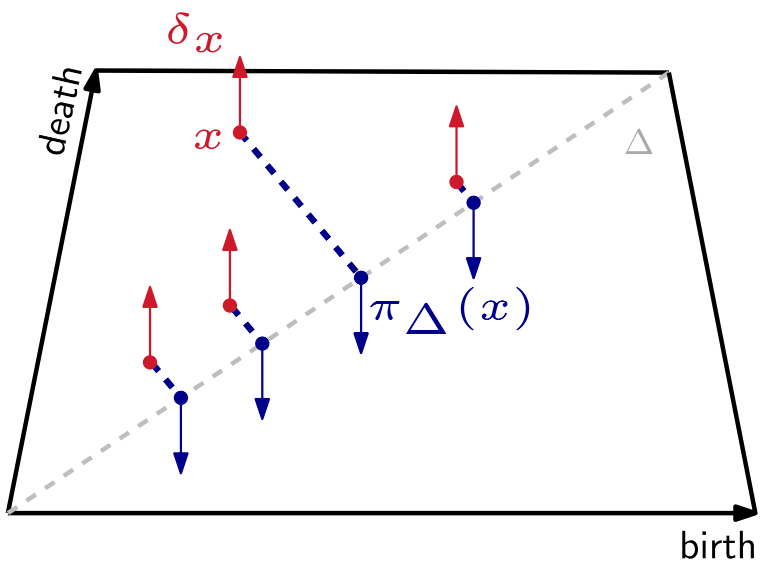
# Sliced Wasserstein kernel

**Def:** Given $\sigma > 0$, for any $\mu, \nu \in \mathcal{M}_+(\mathbb{R}^2)$:

$$k_{SW}(\mu,\nu) := \exp\left(-\frac{SW_1(\mu,\nu)}{2\sigma^2}\right)$$

**Corollary:** [Kolouri, Zou, Rohde](from $SW$ cnsd)
$k_{SW}$ is positive semidefinite.

$\rightarrow$ application to persistence diagrams:




$$D \mapsto \mu_D := \textstyle\sum_{x\in D} \delta_x$$

$$\mapsto \tilde{\mu}_D := \mu_D - \pi_\Delta \# \mu_D$$

$$SW_1(D,D') := \int_{\theta\in\mathcal{S}^1} \|\pi_\theta\#\tilde{\mu}_D - \pi_\theta\#\tilde{\mu}_{D'}\|_K \, d\theta$$

$$k_{SW}(D,D') := \exp\left(-\frac{SW_1(D,D')}{2\sigma^2}\right)$$

# Sliced Wasserstein kernel

**Def:** Given $\sigma > 0$, for any $\mu, \nu \in \mathcal{M}_+(\mathbb{R}^2)$:

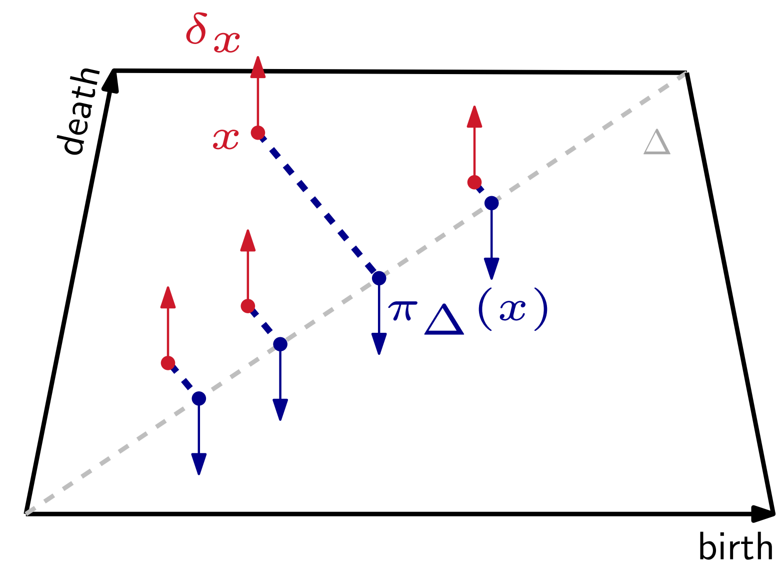
$$k_{SW}(\mu, \nu) := \exp\left(-\frac{SW_1(\mu, \nu)}{2\sigma^2}\right)$$

**Corollary:** [Kolouri, Zou, Rohde](from $SW$ cnsd)
$k_{SW}$ is positive semidefinite.

$\rightarrow$ application to persistence diagrams:

$$D \mapsto \mu_D := \textstyle\sum_{x \in D} \delta_x$$

$$\mapsto \tilde{\mu}_D := \mu_D - \pi_\Delta \# \mu_D$$




$$SW_1(D, D') := \int_{\theta \in \mathcal{S}^1} \|\pi_\theta \# \tilde{\mu}_D - \pi_\theta \# \tilde{\mu}_{D'}\|_K \, d\theta$$

$$k_{SW}(D, D') := \exp\left(-\frac{SW_1(D, D')}{2\sigma^2}\right)$$

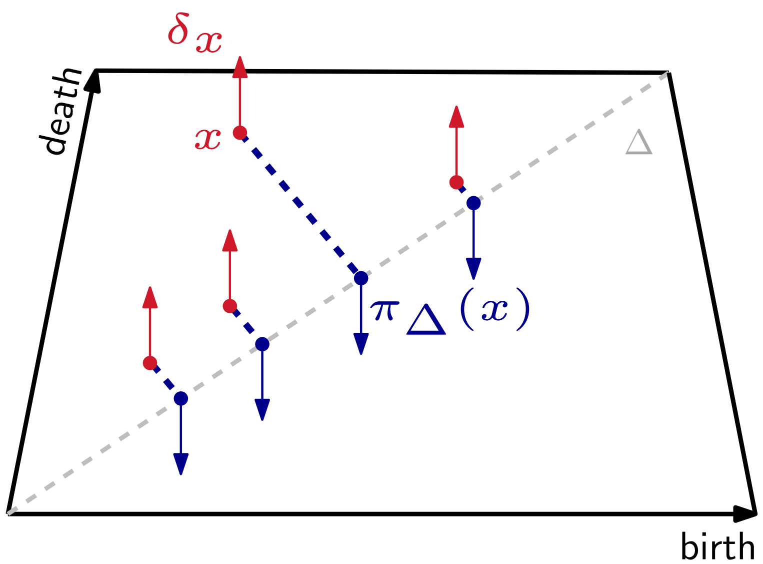
- positive semidefinite
- simple and fast to compute

# Sliced Wasserstein kernel

**Thm.:** [Carrière, Cuturi, O. 2017]
The metrics $d_1$ and $SW_1$ on the space $\mathcal{D}_N$ of persistence diagrams of size bounded by $N$ are strongly equivalent, namely: for $D, D' \in \mathcal{D}_N$,

$$\frac{1}{2+4N(2N-1)}\, d_1(D,D') \;\leq\; SW_1(D,D') \;\leq\; 2\sqrt{2}\, d_1(D,D')$$

$\rightarrow$ application to persistence diagrams:




$$D \mapsto \mu_D := \textstyle\sum_{x\in D} \delta_x$$

$$\mapsto \tilde{\mu}_D := \mu_D - \pi_\Delta \# \mu_D$$

$$SW_1(D,D') := \int_{\theta\in\mathcal{S}^1} \|\pi_\theta\#\tilde{\mu}_D - \pi_\theta\#\tilde{\mu}_{D'}\|_K \, d\theta$$

$$k_{SW}(D,D') := \exp\left(-\frac{SW_1(D,D')}{2\sigma^2}\right)$$

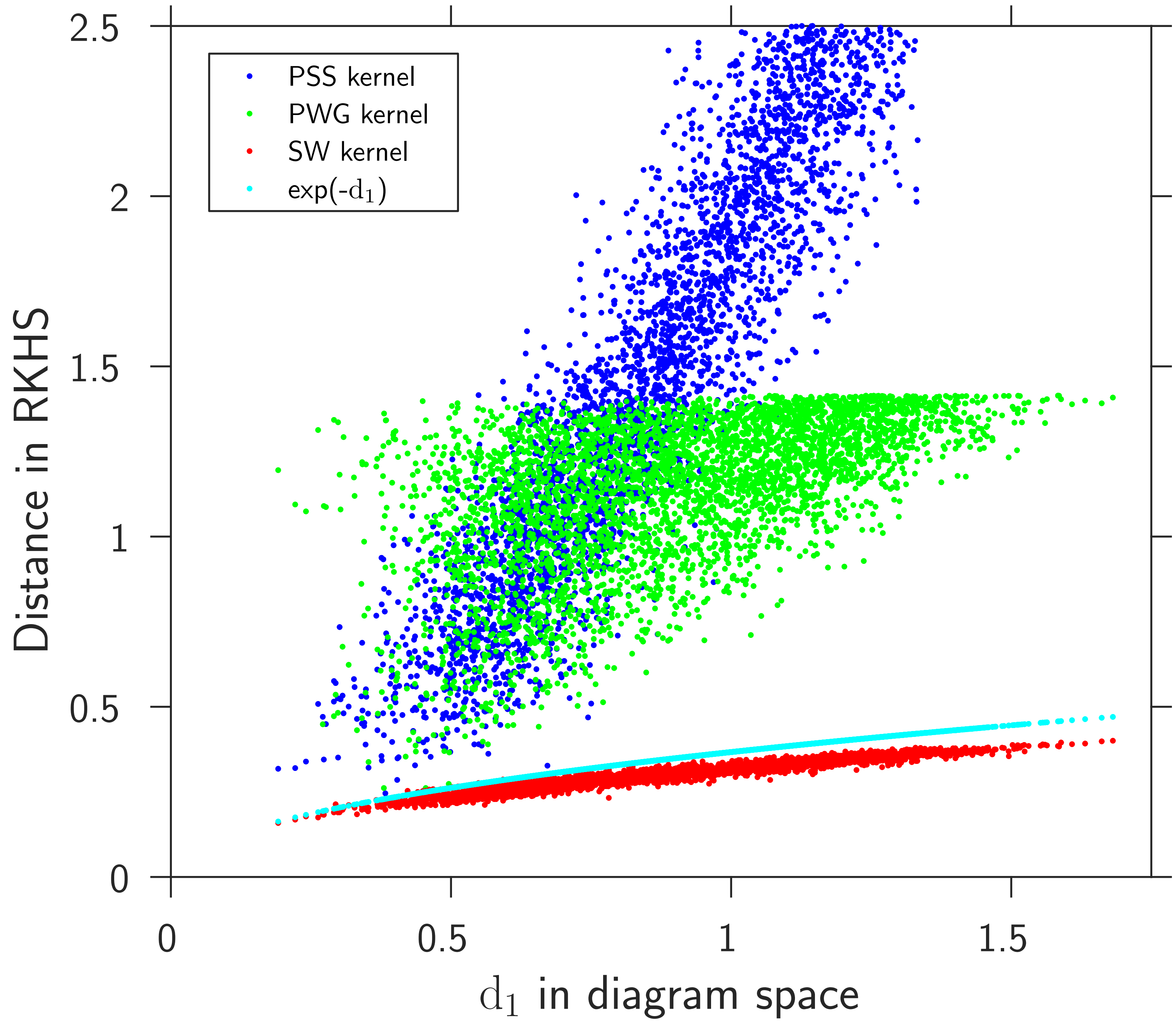
# Sliced Wasserstein kernel

**Thm.:** [Carrière, Cuturi, O. 2017]
The metrics $d_1$ and $SW_1$ on the space $\mathcal{D}_N$ of persistence diagrams of size bounded by $N$ are strongly equivalent, namely: for $D, D' \in \mathcal{D}_N$,

$$\frac{1}{2 + 4N(2N-1)}\, d_1(D, D') \;\leq\; SW_1(D, D') \;\leq\; 2\sqrt{2}\, d_1(D, D')$$

**Corollary:** the feature map $\phi$ associated with $k_{SW}$ is weakly metric-preserving: $\exists g, h$ nonzero except at $0$ such that $g \circ d_1 \leq \|\phi(\cdot) - \phi(\cdot)\|_{\mathcal{H}} \leq h \circ d_1$.

# Metric distortion in practice

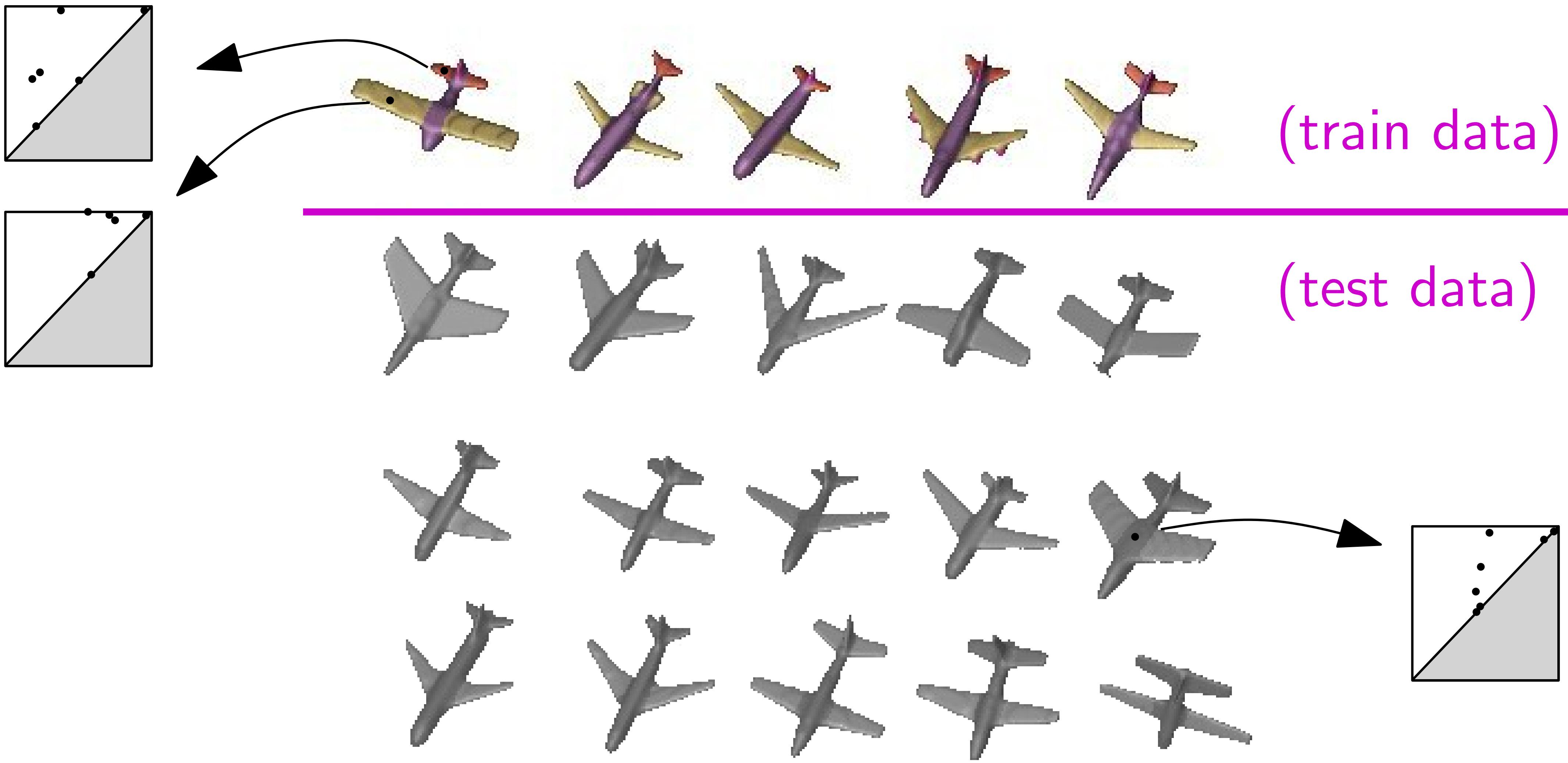


# Application to supervised shape segmentation

**Goal**: segment 3d shapes based on examples

Approach:

- train a (multiclass) classifier on PDs extracted from the training shapes
- apply classifier to PDs extracted from query shape



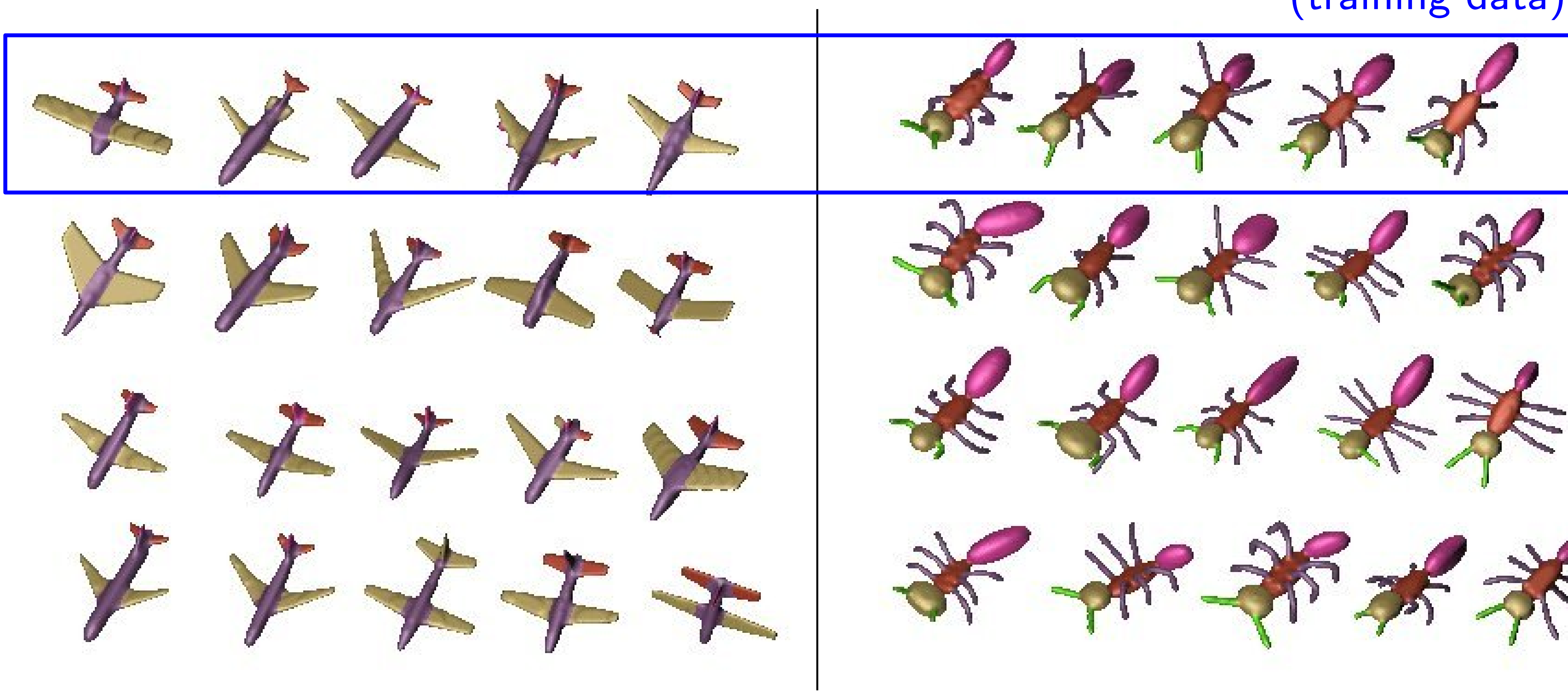
# Application to supervised shape segmentation

**Goal**: segment 3d shapes based on examples

Approach:

- train a (multiclass) classifier on PDs extracted from the training shapes
- apply classifier to PDs extracted from query shape

(training data)



# Application to supervised shape segmentation

**Goal**: segment 3d shapes based on examples

Approach:

- train a (multiclass) classifier on PDs extracted from the training shapes
- apply classifier to PDs extracted from query shape

**Error rates (%):**

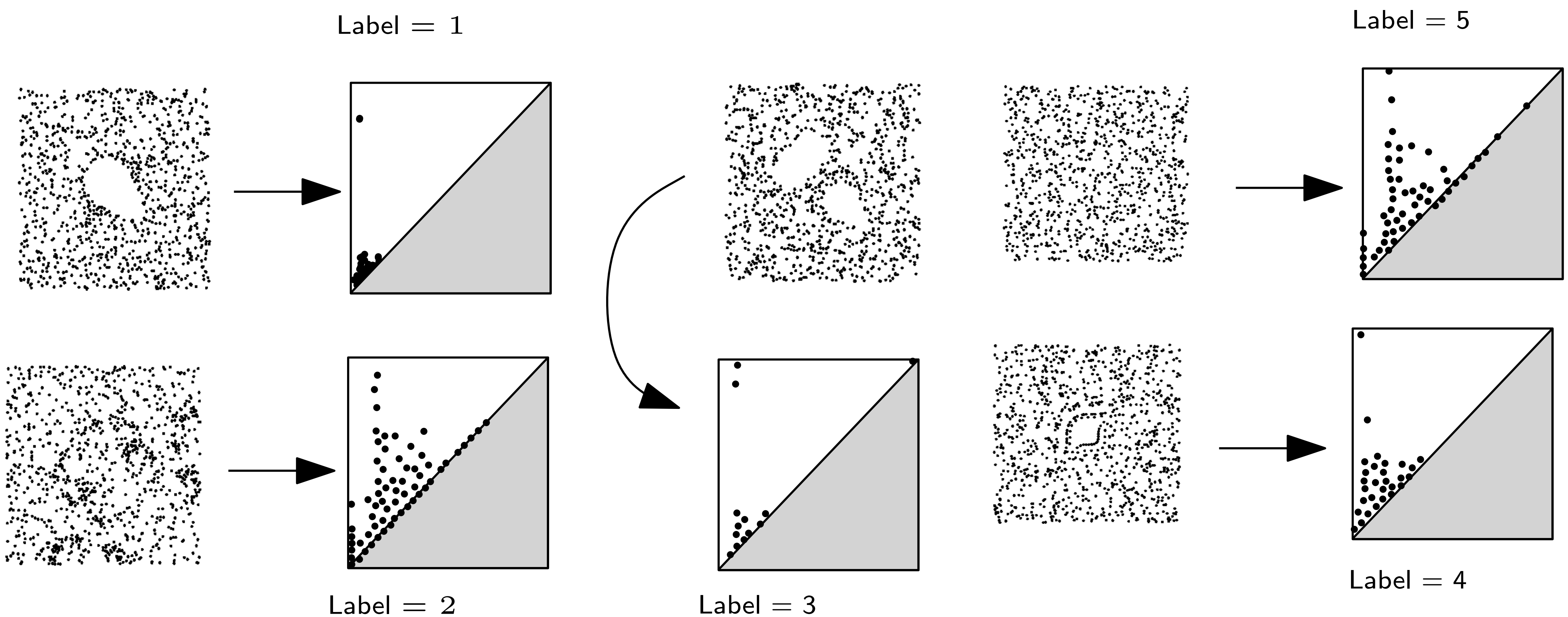
| | TDA | geometry/stats | TDA + geometry/stats |
|---|---|---|---|
| Human | 26.0 | 21.3 | **11.3** |
| Airplane | 27.4 | 18.7 | **9.3** |
| Ant | 7.7 | 9.7 | **1.5** |
| FourLeg | 27.0 | 25.6 | **15.8** |
| Octopus | 14.8 | 5.5 | **3.4** |
| Bird | 28.0 | 24.8 | **13.5** |
| Fish | 20.4 | 20.9 | **7.7** |

# Application to supervised orbits classification

**Goal**: classify orbits of *linked twisted map*, modelling fluid flow dynamics

Orbits described by (depending on parameter $r$):

$$\begin{cases} x_{n+1} = x_n + r\, y_n(1 - y_n) \mod 1 \\ y_{n+1} = y_n + r\, x_{n+1}(1 - x_{n+1}) \mod 1 \end{cases}$$



# Application to supervised orbits classification

**Goal**: classify orbits of *linked twisted map*, modelling fluid flow dynamics

Orbits described by (depending on parameter $r$):

$$\begin{cases} x_{n+1} = x_n + r\, y_n(1 - y_n) \mod 1 \\ y_{n+1} = y_n + r\, x_{n+1}(1 - x_{n+1}) \mod 1 \end{cases}$$

**Accuracies ($\%$) using only TDA descriptors (kernels on barcodes):**

| | $k_{\mathrm{PSS}}$ | $k_{\mathrm{PWG}}$ | $k_{\mathrm{SW}}$ |
|---|---|---|---|
| Orbit | $64.0 \pm 0.0$ | $78.7 \pm 0.0$ | $\mathbf{83.7} \pm 1.1$ |

(PDs as discrete measures)

**Running times (in seconds on $N$-sized parameter space from 100 orbits):**

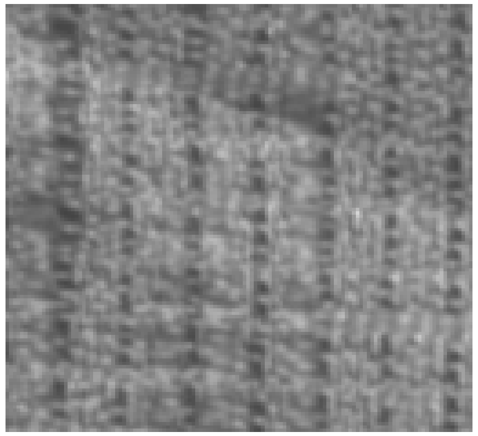
| | $k_{\mathrm{PSS}}$ | $k_{\mathrm{PWG}}$ | $k_{\mathrm{SW}}$ |
|---|---|---|---|
| Orbit | $N \times 9183.4 \pm 65.6$ | $N \times 69.2 \pm 0.9$ | $385.8 \pm 0.2 + NC$ |

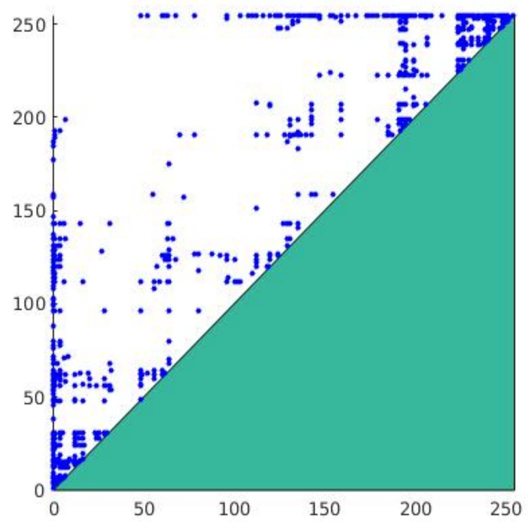
# Application to supervised texture classification

**Goal**: classify textures from the OUTEX00000 database [Ojala et al. 2002]

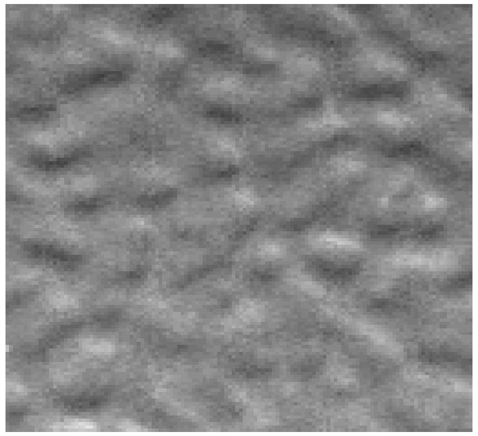
Textures described by CLBP (Compound Local Binary Pattern) [Guo et al. 2010]

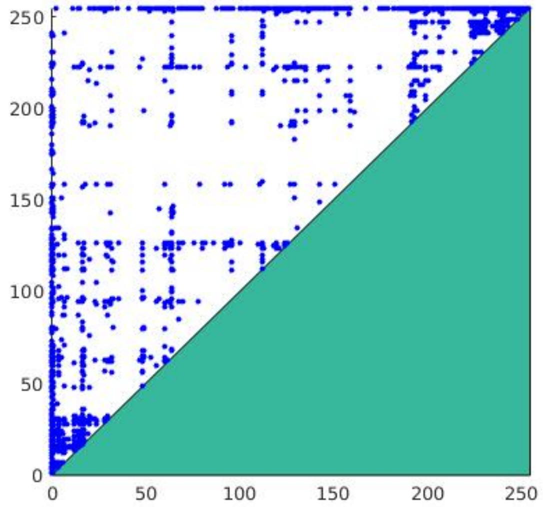
$\rightarrow$ apply degree-0 persistence on 1st sign component



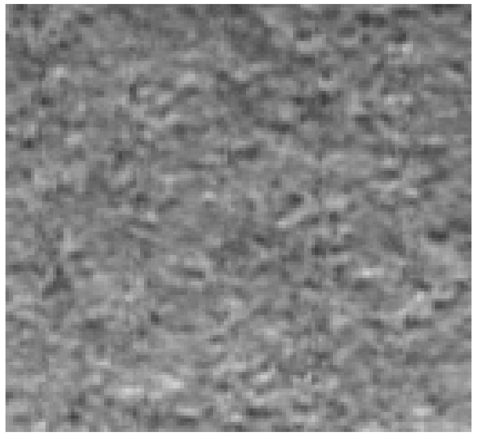


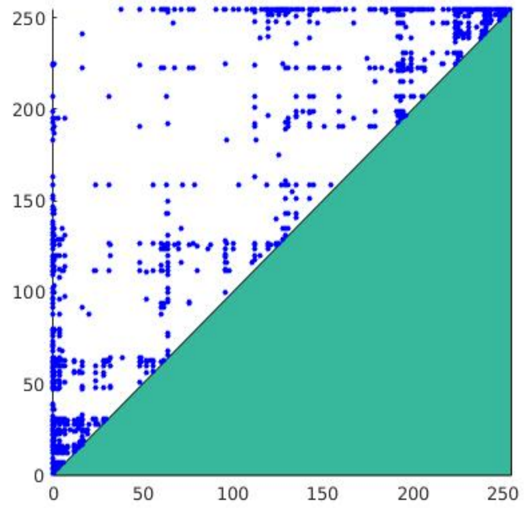

Label = Canvas






Label = Carpet






Label = Tile

# Application to supervised texture classification

**Goal**: classify textures from the OUTEX00000 database [Ojala et al. 2002]

Textures described by CLBP (Compound Local Binary Pattern) [Guo et al. 2010]

$\to$ apply degree-0 persistence on 1st sign component

**Accuracies (%) using only TDA descriptors (kernels on barcodes):**

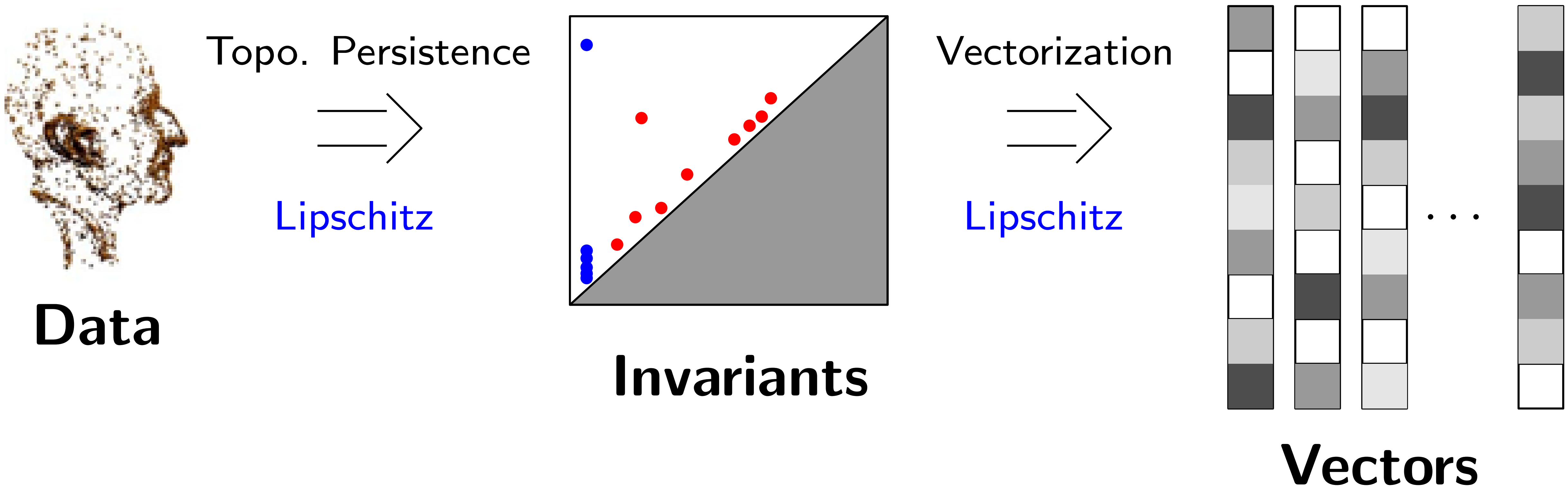
| | $k_{\mathrm{PSS}}$ | $k_{\mathrm{PWG}}$ | $k_{\mathrm{SW}}$ |
|---|---|---|---|
| Orbit | $\mathbf{98.7} \pm 0.06$ | $96.7 \pm 0.4$ | $96.1 \pm 0.1$ |

(PDs as discrete measures)

**Running times (in seconds on $N$-sized parameter space from 100 orbits):**

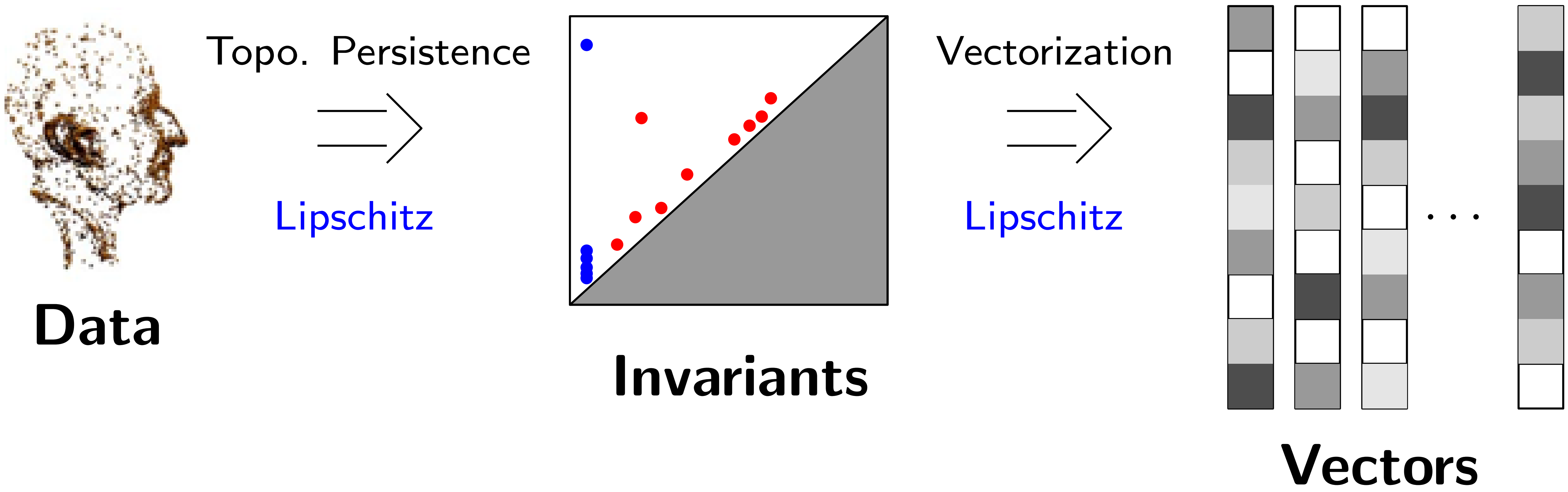
| | $k_{\mathrm{PSS}}$ | $k_{\mathrm{PWG}}$ | $k_{\mathrm{SW}}$ |
|---|---|---|---|
| Orbit | $N \times 10337.4 \pm 140.5$ | $N \times 45.9 \pm 0.6$ | $126.4 \pm 0.2 + NC$ |

# Back to the TDA pipeline




**Thm** (Rademacher): pipeline is differentiable almost everywhere

# Back to the TDA pipeline




**Thm** (Rademacher): pipeline is differentiable almost everywhere

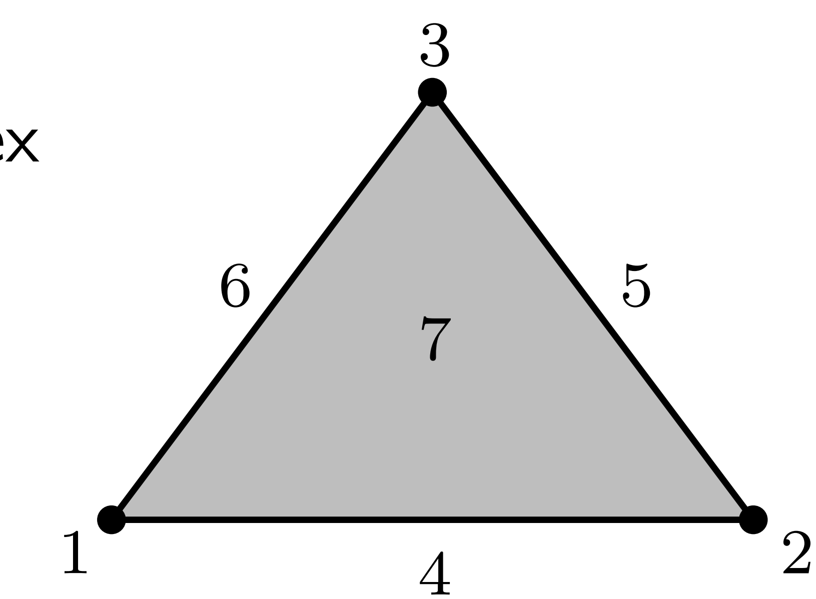
---

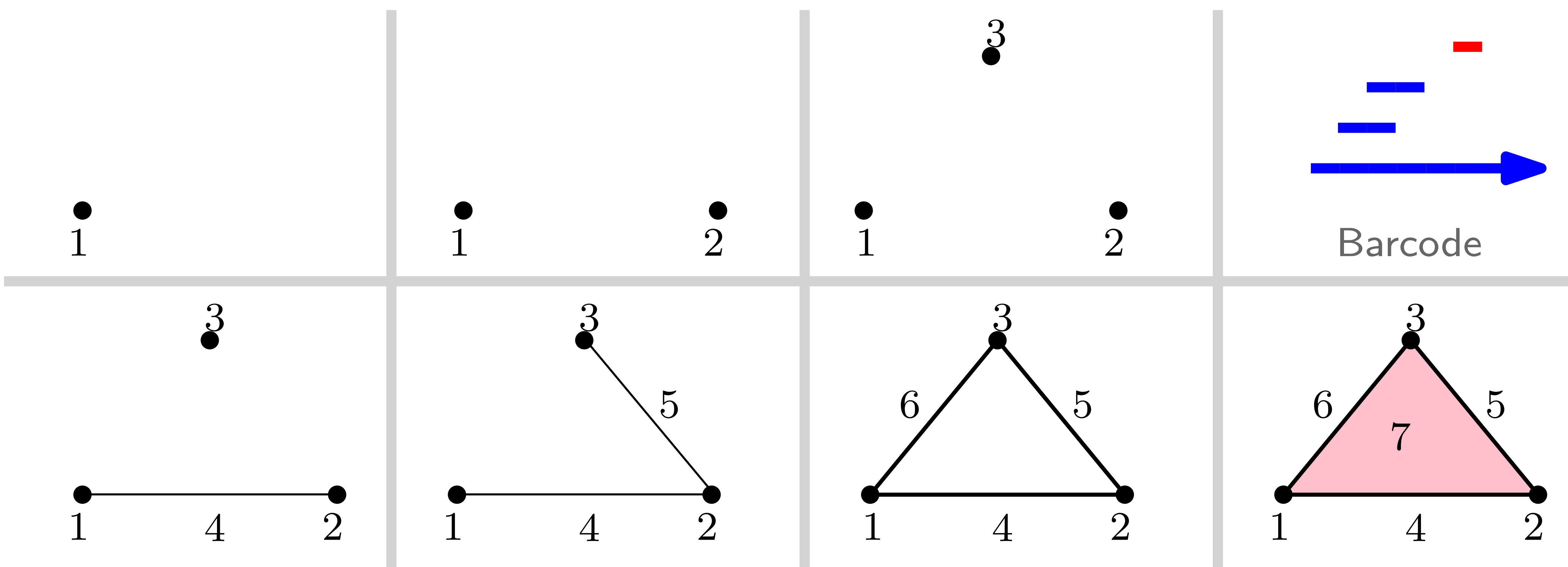
**Questions:**

- class of differentiability?
- derivatives? chain rule?
- non-differentiablity set?

# The persistence algorithm

**Input:** $f\colon X \to \mathbb{R}$ where $X$ finite simplicial complex and $f(\tau) \leq f(\sigma)$ for all faces $\tau \subseteq \sigma \in X$

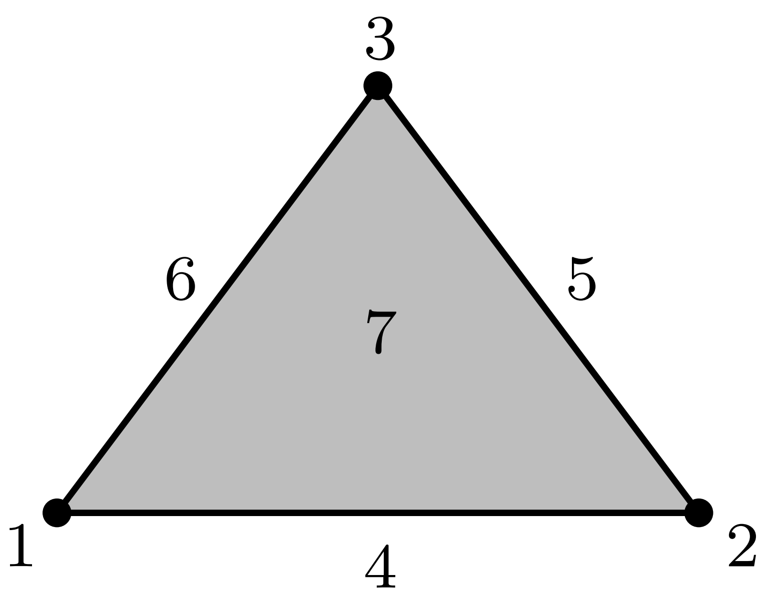


# The persistence algorithm

**Input:** $f : X \to \mathbb{R}$ where $X$ finite simplicial complex
and $f(\tau) \leq f(\sigma)$ for all faces $\tau \subseteq \sigma \in X$

**Output:** boundary matrix

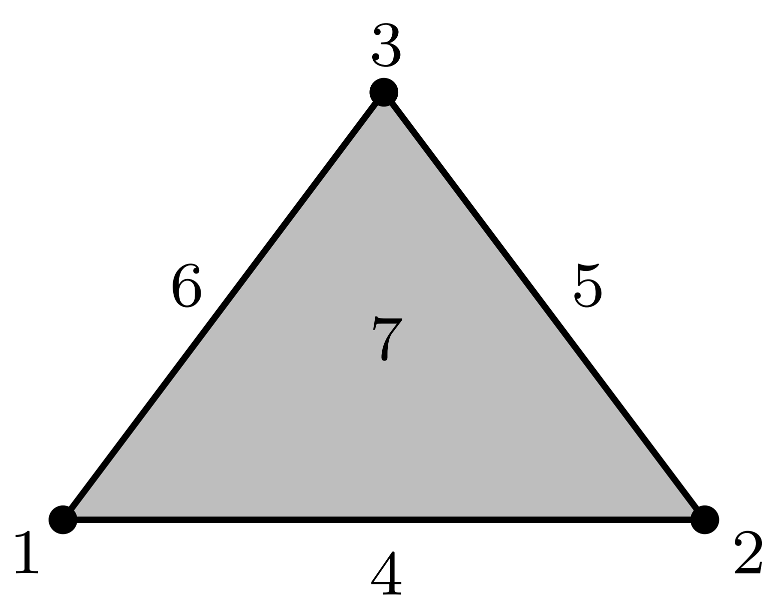



| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 1 | | | | $*$ | | $*$ | |
| 2 | | | | $*$ | $*$ | | |
| 3 | | | | | $*$ | $*$ | |
| 4 | | | | | | | $*$ |
| 5 | | | | | | | $*$ |
| 6 | | | | | | | $*$ |
| 7 | | | | | | | |

# The persistence algorithm

**Input:** $f : X \to \mathbb{R}$ where $X$ finite simplicial complex
and $f(\tau) \leq f(\sigma)$ for all faces $\tau \subseteq \sigma \in X$

**Output:** boundary matrix in column-echelon form



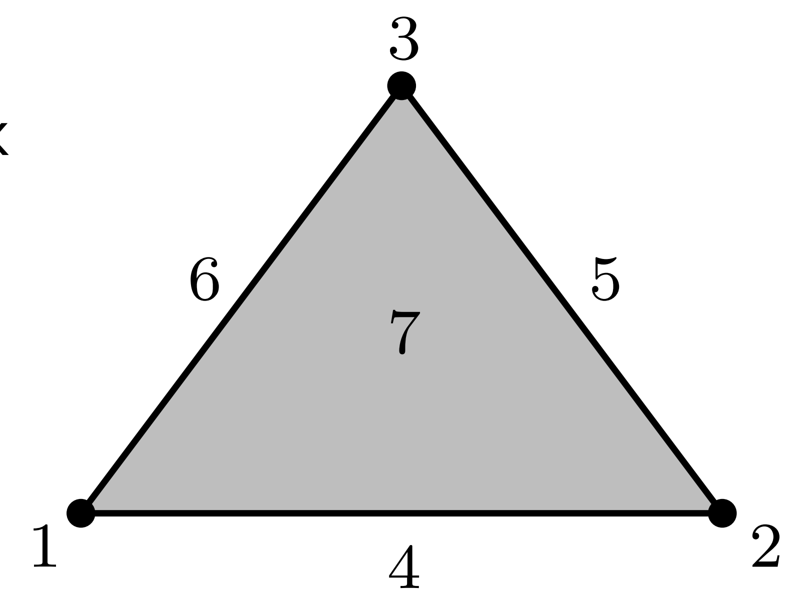

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 1 | | | | $*$ | | $*$ | |
| 2 | | | | $*$ | $*$ | | |
| 3 | | | | | $*$ | $*$ | |
| 4 | | | | | | | $*$ |
| 5 | | | | | | | $*$ |
| 6 | | | | | | | $*$ |
| 7 | | | | | | | |

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 1 | | | | $*$ | | | |
| 2 | | | | 1 | $*$ | | |
| 3 | | | | | 1 | | |
| 4 | | | | | | | $*$ |
| 5 | | | | | | | $*$ |
| 6 | | | | | | | 1 |
| 7 | | | | | | | |

# The persistence algorithm

**Input:** $f : X \to \mathbb{R}$ where $X$ finite simplicial complex
and $f(\tau) \leq f(\sigma)$ for all faces $\tau \subseteq \sigma \in X$

**Output:** boundary matrix in column-echelon form




pivots pair up simplices → finite intervals: $[2,4)$, $[3,5)$, $[6,7)$

unpaired simplices → infinite intervals: $[1,+\infty)$

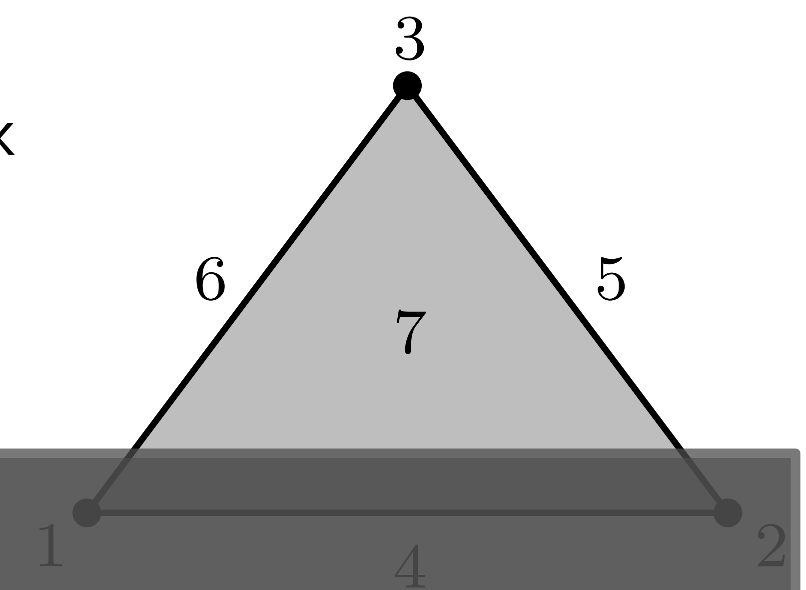
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 1 | | | | * | | * | |
| 2 | | | | * | * | | |
| 3 | | | | | * | * | |
| 4 | | | | | | | * |
| 5 | | | | | | | * |
| 6 | | | | | | | * |
| 7 | | | | | | | |

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 1 | | | | * | | | |
| 2 | | | | 1 | * | | |
| 3 | | | | | 1 | | |
| 4 | | | | | | | * |
| 5 | | | | | | | * |
| 6 | | | | | | | 1 |
| 7 | | | | | | | |

# The persistence algorithm

**Input:** $f : X \to \mathbb{R}$ where $X$ finite simplicial complex
and $f(\tau) \le f(\sigma)$ for all faces $\tau \subseteq \sigma \in X$

**Output:** boundary matrix in column-echelon form




pivots pair up simplices $\to$ finite intervals: $[2,4)$, $[3,5)$, $[6,7)$

unpaired simplices $\to$ infinite intervals: $[1,+\infty)$

**Key observations:**

- pairing depends only on simplex (pre-)order induced by $f$
- under fixed pairing, barcode endpoints depend linearly on $f$-values

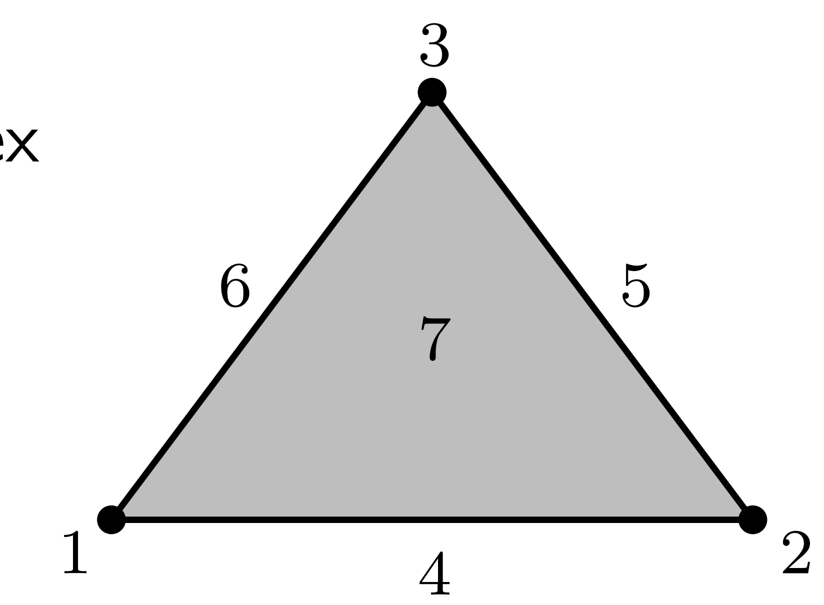
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 1 | | | | | | * | |
| 2 | | | | * | * | | |
| 3 | | | | | * | * | |
| 4 | | | | | | | * |
| 5 | | | | | | | * |
| 6 | | | | | | | * |
| 7 | | | | | | | |

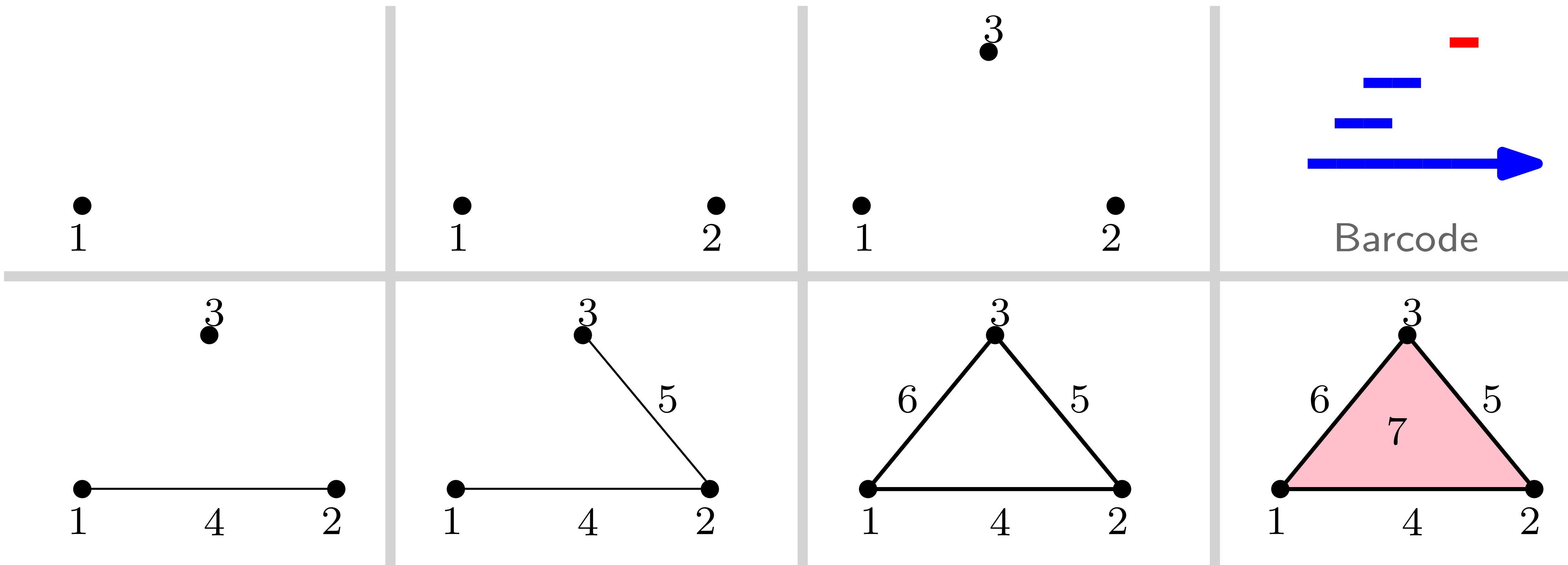
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| $\boxed{1}$ | | | | * | | | |
| 2 | | | | 1 | * | | |
| 3 | | | | | 1 | | |
| 4 | | | | | | | * |
| 5 | | | | | | | * |
| 6 | | | | | | | 1 |
| 7 | | | | | | | |

# The persistence algorithm

**Input:** $f\colon X \to \mathbb{R}$ where $X$ finite simplicial complex
and $f(\tau) \leq f(\sigma)$ for all faces $\tau \subseteq \sigma \in X$

**Output:** boundary matrix in column-echelon form

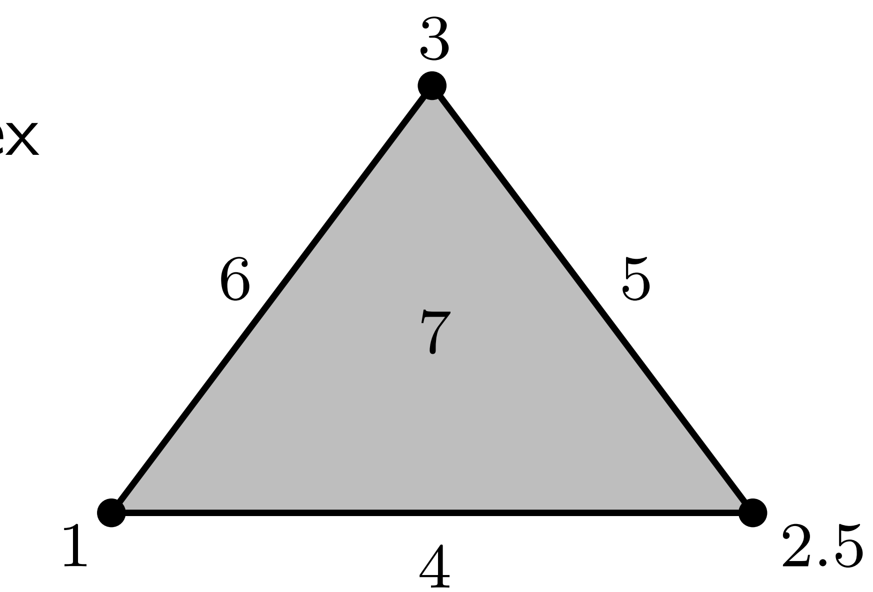


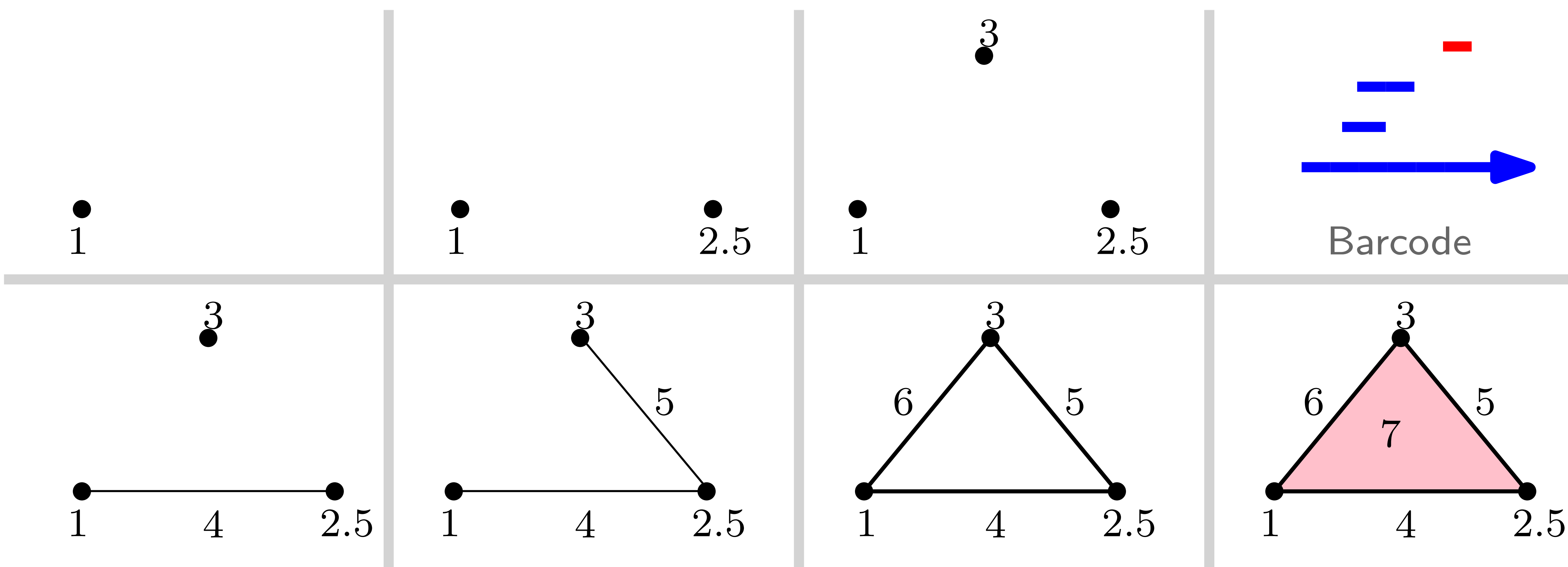



# The persistence algorithm

**Input:** $f\colon X \to \mathbb{R}$ where $X$ finite simplicial complex
and $f(\tau) \leq f(\sigma)$ for all faces $\tau \subseteq \sigma \in X$

**Output:** boundary matrix in column-echelon form



# Mathematical formulation [Leygonie et al. '21] [Carrière et al. '21]

$$\begin{tikzcd} & (\mathbb{R} \cup \{\infty\})^{2m} \arrow[d, shift right, "q"'] \\ \mathrm{Filter}(X) \arrow[ur, dotted] \arrow[r, "\mathrm{Pers}"'] \arrow[d, phantom, "\rotatebox{90}{$\subseteq$}"] & \mathrm{Bar} \arrow[u, shift right, "p"'] \\ \mathbb{R}^m & \end{tikzcd}$$

$X$: fixed simplicial complex with $m$ simplices

Filter$(X)$: affine cone of filter functions on $X$

$\mathrm{Pers}$: persistence map (algorithm)

Bar: space of persistence barcodes / diagrams

$p$: lexicographic ordering of bars / $q$: pairing of consecutive coordinates

$q \circ p = \mathrm{id}_{\mathsf{Bar}}$

**Prop:** $p \circ \mathrm{Pers}$ is piecewise affine, with an affine underlying partition of Filter$(X)$.

# Mathematical formulation [Leygonie et al. '21] [Carrière et al. '21]

$$\begin{tikzcd}
& & (\mathbb{R} \cup \{\infty\})^{2m} \arrow[d, shift right=1ex, "q"'] \\
\mathcal{M} \arrow[r, "F"'] \arrow[urr, dotted] & \mathsf{Filter}(X) \arrow[r, "\mathrm{Pers}"'] \arrow[ur, dotted] \arrow[d, phantom, sloped, "\subseteq"] & \mathsf{Bar} \arrow[u, shift right=1ex, "p"'] \\
& \mathbb{R}^m &
\end{tikzcd}$$

$X$: fixed simplicial complex with $m$ simplices

Filter$(X)$: affine cone of filter functions on $X$

$\mathrm{Pers}$: persistence map (algorithm)

Bar: space of persistence barcodes / diagrams

$p$: lexicographic ordering of bars / $q$: pairing of consecutive coordinates

$F$: parametrized family of filter functions

**Prop:** $p \circ \mathrm{Pers}$ is piecewise affine, with an affine underlying partition of Filter$(X)$.

**Consequence:** if $F$ is semialgebraic or subanalytic, then so is $p \circ \mathrm{Pers} \circ F$.

# Mathematical formulation [Leygonie et al. '21] [Carrière et al. '21]

$$\begin{tikzcd}
& & (\mathbb{R}\cup\{\infty\})^{2m} \arrow[d, shift right, "q"'] \arrow[dr, dotted] \arrow[drr, dotted] & & \\
\mathcal{M} \arrow[r, "F"'] \arrow[urr, dotted] & \mathsf{Filter}(X) \arrow[r, "\mathrm{Pers}"'] \arrow[ur, dotted] \arrow[d, phantom, sloped, "\subseteq"] & \mathsf{Bar} \arrow[u, shift right, "p"'] \arrow[r, "V"'] & \mathbb{R}^d \arrow[r, "\mathcal{L}"'] & \mathbb{R} \\
& \mathbb{R}^m & & &
\end{tikzcd}$$

$X$: fixed simplicial complex with $m$ simplices

Filter$(X)$: affine cone of filter functions on $X$

$\mathrm{Pers}$: persistence map (algorithm)

Bar: space of persistence barcodes / diagrams

**Prop:** $p \circ \mathrm{Pers}$ is piecewise affine, with an affine underlying partition of Filter$(X)$.

**Consequence:** if $\mathcal{L} \circ V \circ q$ is also semialgebraic or subanalytic, then so is $\mathcal{L} \circ V \circ \mathrm{Pers} \circ F$

$p$: lexicographic ordering of bars / $q$: pairing of consecutive coordinates

$F$: parametrized family of filter functions $V$: vectorization $\mathcal{L}$: loss function

# Mathematical formulation [Leygonie et al. '21] [Carrière et al. '21]

$$\begin{tikzcd}
 & & (\mathbb{R}\cup\{\infty\})^{2m} \arrow[d, shift right, "q"'] \arrow[dr, dotted] \arrow[drr, dotted] & & \\
\mathcal{M} \arrow[r, "F"'] \arrow[urr, dotted] & \mathsf{Filter}(X) \arrow[r, "\mathrm{Pers}"'] \arrow[ur, dotted] \arrow[d, phantom, sloped, "\subseteq"] & \mathsf{Bar} \arrow[u, shift right, "p"'] \arrow[r, "V"'] & \mathbb{R}^d \arrow[r, "\mathcal{L}"'] & \mathbb{R} \\
 & \mathbb{R}^m & & &
\end{tikzcd}$$

$X$: fixed simplicial complex with $m$ simplices

Filter$(X)$: affine cone of filter functions on $X$

$\mathrm{Pers}$: persistence map (algorithm)

Bar: space of persistence barcodes / diagrams

$p$: lexicographic ordering of bars / $q$: pairin

$F$: parametrized family of filter functions $V$: vectorization $\mathcal{L}$: loss function

**Prop:** $p \circ \mathrm{Pers}$ is piecewise affine, with an affine underlying partition of Filter$(X)$.

**Consequence:** if $\mathcal{L} \circ V \circ q$ is also semialgebraic or subanalytic, then so is $\mathcal{L} \circ V \circ \mathrm{Pers} \circ F$, with chain rule on top strata:

$$\nabla_m (\mathcal{L} \circ V \circ \mathrm{Pers} \circ F) =$$

$$\nabla_{p \circ \mathrm{Pers} \circ F(m)} (\mathcal{L} \circ V \circ q)\, \mathbf{J}_m (p \circ \mathrm{Pers} \circ F)$$

# Application to inverse problems [Gameiro et al. '16]

**Point cloud continuation**

**Goal:** given a labeled point cloud $P = \{p_1, \cdots, p_n\} \subset \mathbb{R}^d$ and its corresponding barcode/diagram $D$, describe changes in $P$ under small perturbations of $D$.


$P$ Rips filtration $D$

$\cap$

Bar

$\mathcal{M} = \mathbb{R}^{n \times d}$

$$\begin{pmatrix} X = 2^P \setminus \{\emptyset\} \\ \forall \emptyset \neq \sigma \subseteq P,\ f(\sigma) = \max_{p_i, p_j \in \sigma} \|p_i - p_j\| \end{pmatrix}$$

# Application to inverse problems [Gameiro et al. '16]

## Point cloud continuation

**Goal:** given a labeled point cloud $P = \{p_1, \cdots, p_n\} \subset \mathbb{R}^d$ and its corresponding barcode/diagram $D$, describe changes in $P$ under small perturbations of $D$.


$P$ Rips filtration $D$

$\mathcal{M} = \mathbb{R}^{n\times d}$

$$\left( \begin{array}{c} X = 2^P \setminus \{\emptyset\} \\ \forall \emptyset \neq \sigma \subseteq P,\ f(\sigma) = \max\limits_{p_i, p_j \in \sigma} \|p_i - p_j\| \end{array} \right)$$

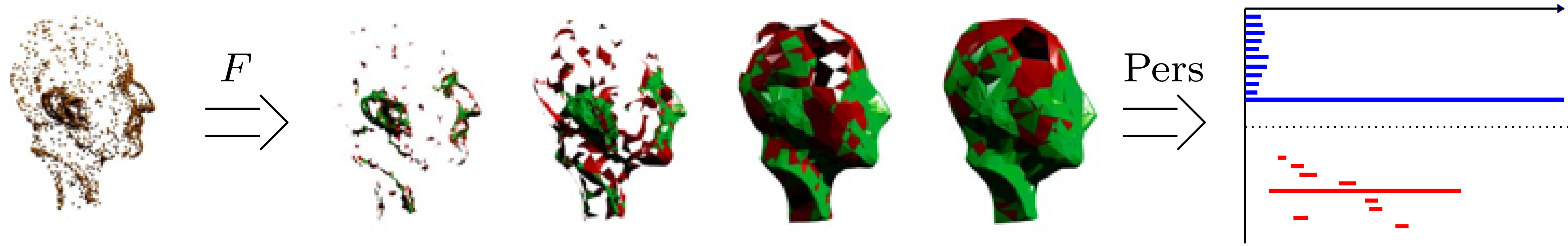
$D$
$\cap$
Bar

- [from 2016] order on $X$ induced by $f$ is stable when $P$ is *generic* (all distances differ)

# Application to inverse problems [Gameiro et al. '16]

**Point cloud continuation**

**Goal:** given a labeled point cloud $P = \{p_1, \cdots, p_n\} \subset \mathbb{R}^d$ and its corresponding barcode/diagram $D$, describe changes in $P$ under small perturbations of $D$.




$P$    Rips filtration    $D$

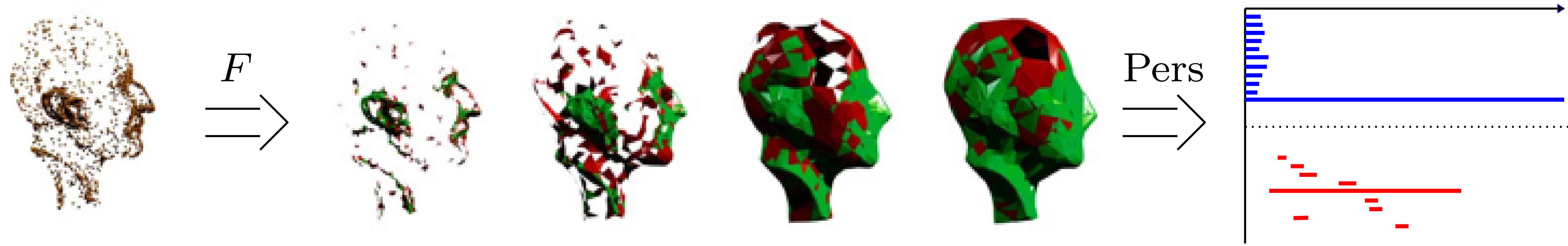
$$\mathcal{M} = \mathbb{R}^{n\times d} \qquad \begin{pmatrix} X = 2^P \setminus \{\emptyset\} \\ \forall \emptyset \neq \sigma \subseteq P,\ f(\sigma) = \max_{p_i, p_j \in \sigma} \|p_i - p_j\| \end{pmatrix} \qquad \begin{matrix} D \\ \Cap \\ \mathrm{Bar} \end{matrix}$$

- [from 2016] order on $X$ induced by $f$ is stable when $P$ is *generic* (all distances differ)
- [from 2021] $p \circ \mathrm{Pers} \circ F$ is semialgebraic, and genericity $\Rightarrow$ $P \in$ top-dimensional stratum

# Application to inverse problems [Gameiro et al. '16]

**Point cloud continuation**

**Goal:** given a labeled point cloud $P = \{p_1, \cdots, p_n\} \subset \mathbb{R}^d$ and its corresponding barcode/diagram $D$, describe changes in $P$ under small perturbations of $D$.




$P$ Rips filtration $D$

$$\mathcal{M} = \mathbb{R}^{n \times d} \qquad \begin{pmatrix} X = 2^P \setminus \{\emptyset\} \\ \forall \emptyset \neq \sigma \subseteq P,\ f(\sigma) = \max_{p_i, p_j \in \sigma} \|p_i - p_j\| \end{pmatrix} \qquad \begin{matrix} D \\ \cap \\ \mathrm{Bar} \end{matrix}$$

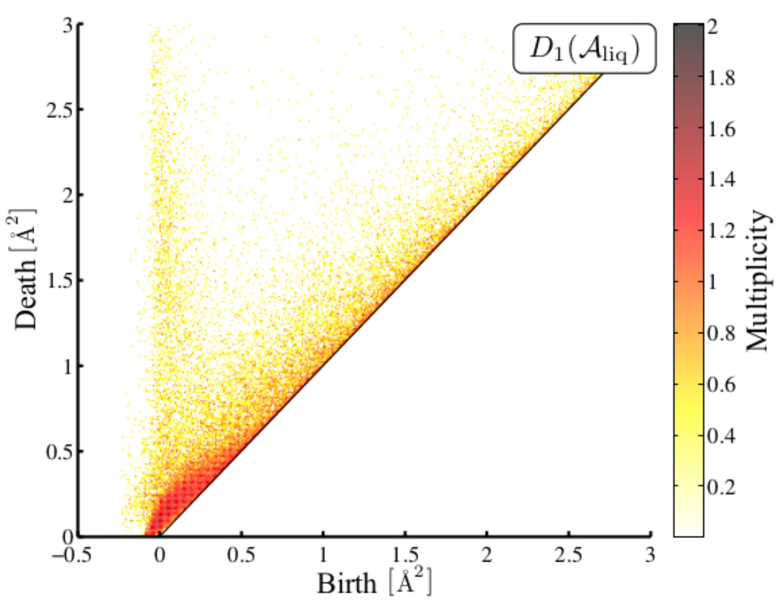
- [from 2016] order on $X$ induced by $f$ is stable when $P$ is *generic* (all distances differ)
- [from 2021] $p \circ \mathrm{Pers} \circ F$ is semialgebraic, and genericity $\Rightarrow$ $P \in$ top-dimensional stratum
- apply inverse function theorem to $p \circ \mathrm{Pers} \circ F$

# Application to inverse problems [Gameiro et al. '16]

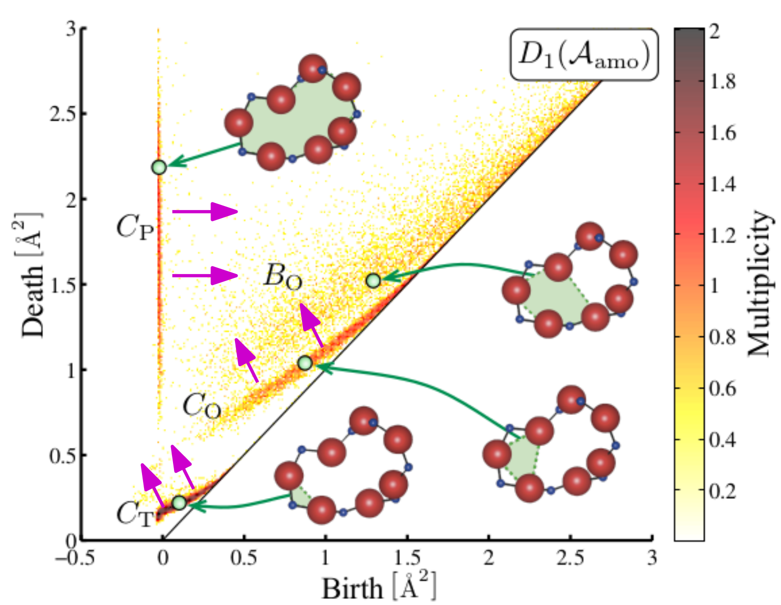
**Point cloud continuation**

**Goal:** given a labeled point cloud $P = \{p_1, \cdots, p_n\} \subset \mathbb{R}^d$ and its corresponding barcode/diagram $D$, describe changes in $P$ under small perturbations of $D$.

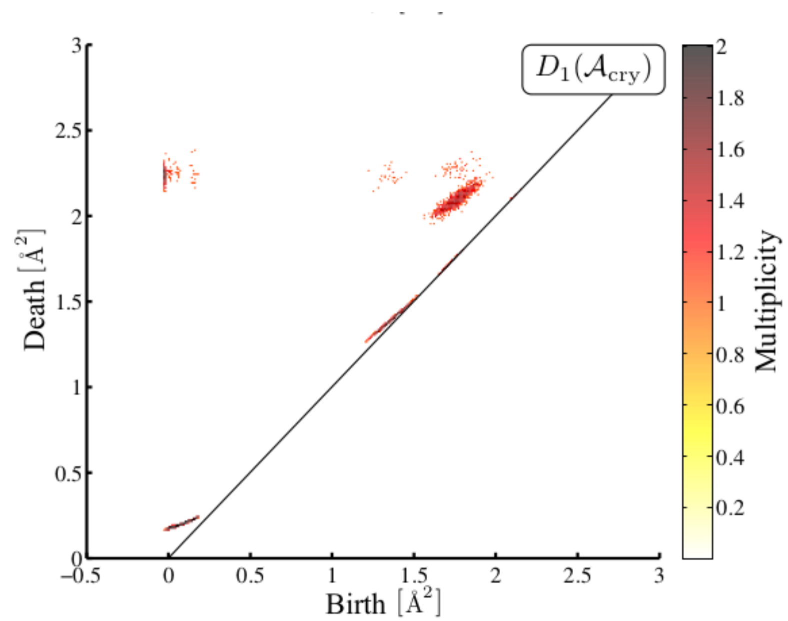
- application to the study of the rigidity of glass [Hiraoka et al. '16]




liquid silica




amorphous silica

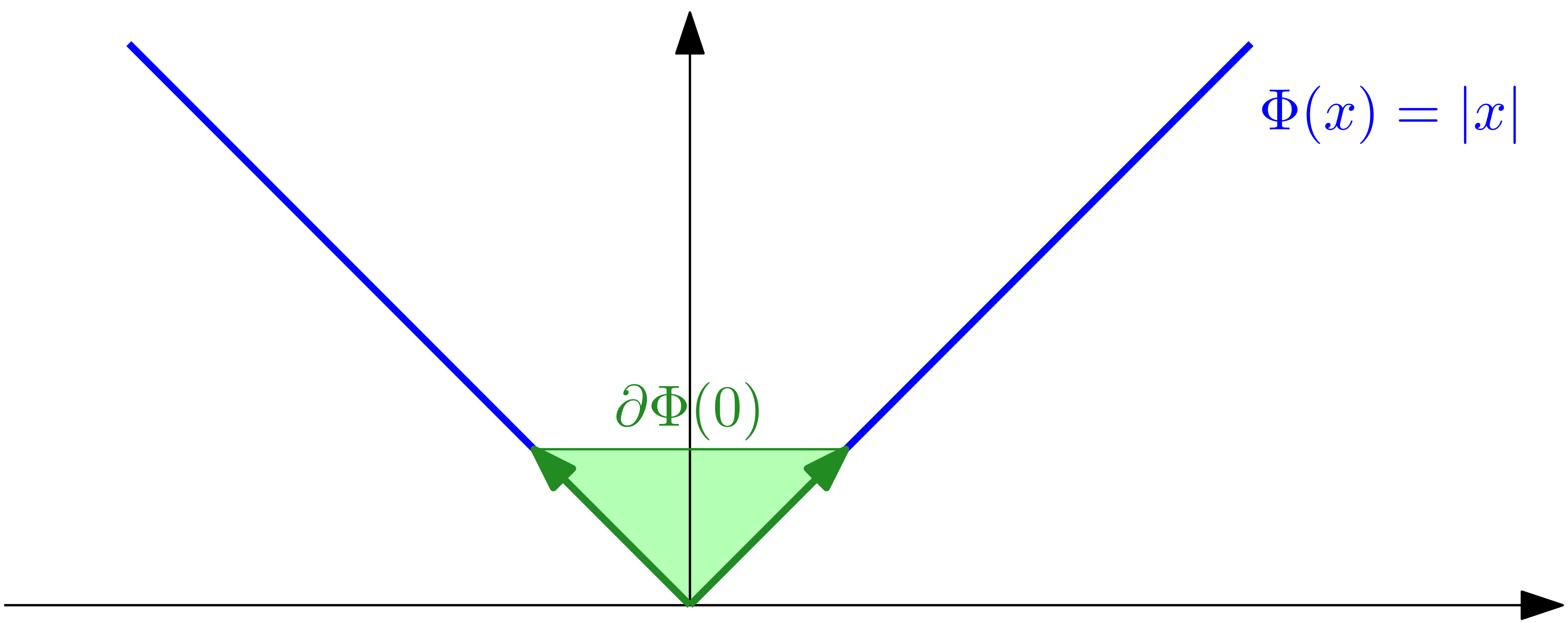



crystalline silica

# Towards nonsmooth optimization

**Prop:** When $\Phi = \mathcal{L} \circ V \circ \mathrm{Pers} \circ F \colon \mathcal{M} \to \mathbb{R}$ is definable (e.g. semialgebraic or subanalytic), it has a well-defined *Clarke subdifferential*:

$$\partial\Phi(x) := \mathrm{Conv}\{\lim_{x' \to x} \nabla\Phi(x') \mid \Phi \text{ differentiable at } x'\}.$$



# Towards nonsmooth optimization

**Prop:** When $\Phi = \mathcal{L} \circ V \circ \mathrm{Pers} \circ F \colon \mathcal{M} \to \mathbb{R}$ is definable (e.g. semialgebraic or subanalytic), it has a well-defined *Clarke subdifferential*:

$$\partial\Phi(x) := \mathrm{Conv}\{\lim_{x' \to x} \nabla\Phi(x') \mid \Phi \text{ differentiable at } x'\}.$$

**Stochastic subgradient descent step:**

$$\overset{\text{iterates}}{x_{k+1}} := \overset{\text{iterates}}{x_k} - \overset{\text{learning rate}}{\alpha_k}(g_k + \overset{\text{centered noise}}{\zeta_k}),$$

where $g_k \in \partial\Phi(x_k)$ (subgradient).

# Towards nonsmooth optimization

**Prop:** When $\Phi = \mathcal{L} \circ V \circ \mathrm{Pers} \circ F \colon \mathcal{M} \to \mathbb{R}$ is definable (e.g. semialgebraic or subanalytic), it has a well-defined *Clarke subdifferential*:

$$\partial\Phi(x) := \mathrm{Conv}\{\lim_{x' \to x} \nabla\Phi(x') \mid \Phi \text{ differentiable at } x'\}.$$

**Stochastic subgradient descent step:**

learning rate centered noise
iterates

$$x_{k+1} := x_k - \alpha_k(g_k + \zeta_k),$$

where $g_k \in \partial\Phi(x_k)$ (subgradient).

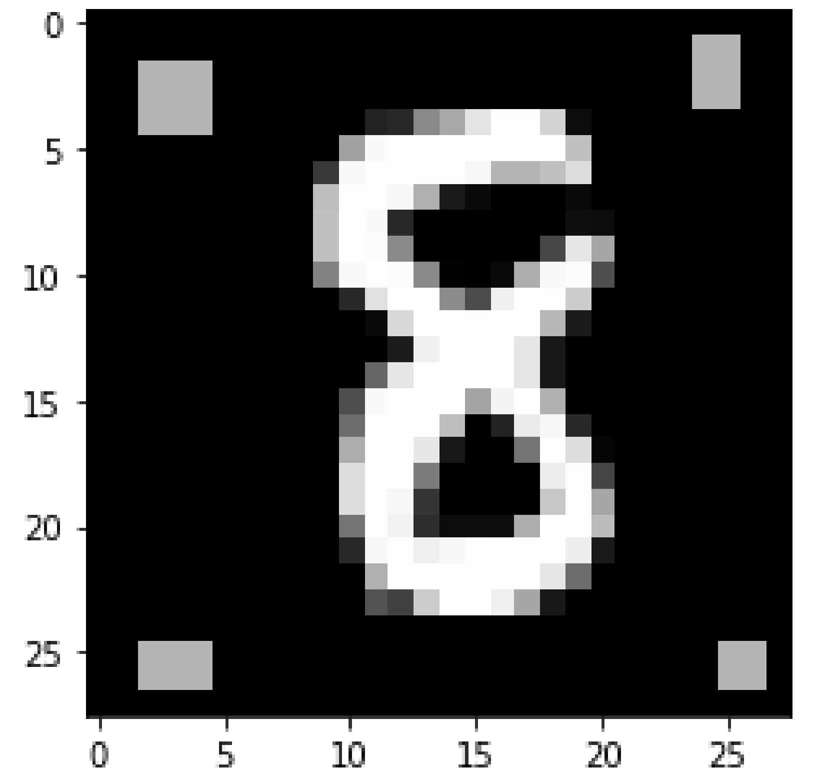
**Thm:** [Davis et al. '20]

Suppose $\Phi$ is definable (e.g. semiagebraic or subanalytic) and locally Lipschitz. Then, under standard conditions on the parameters, almost surely the limit points of the iterates of stochastic subgradient descent are critical for $\Phi$ and the sequence $\{\Phi(x_k)\}_k$ converges.

# Example: image binarization [Carrière et al. '21]

**Input:** greyscaled image $I\colon \{1,\cdots,n\}^2 \to [0,1]$.

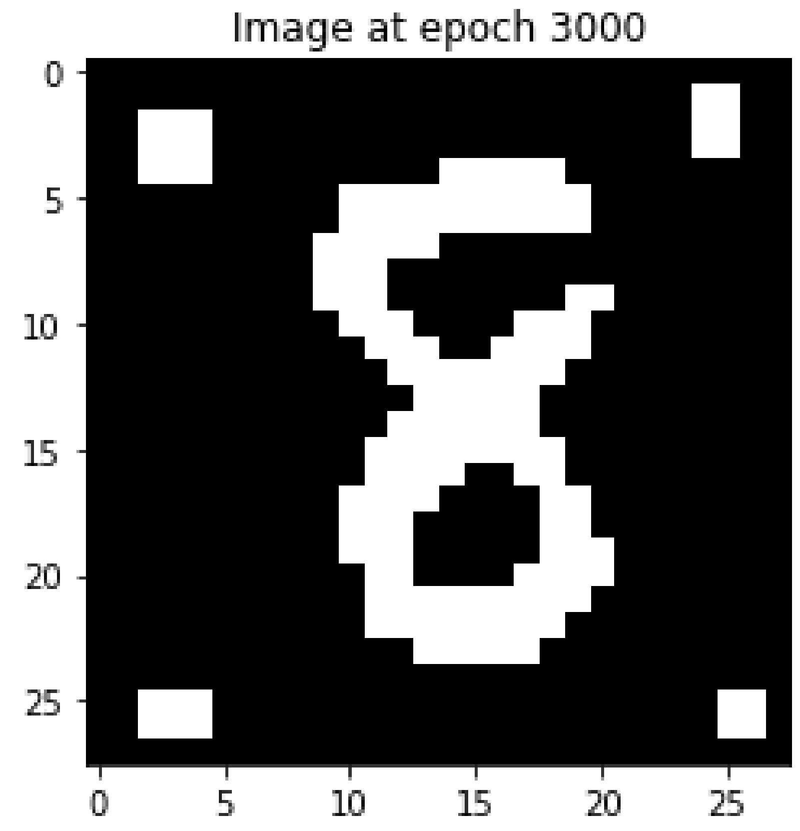
**Output:** image $J : \{1,\cdots,n\}^2 \to \{0,1\}$



# Example: image binarization [Carrière et al. '21]

**Input:** greyscaled image $I\colon \{1,\cdots,n\}^2 \to [0,1]$.

**Output:** image $J : \{1,\cdots,n\}^2 \to \{0,1\}$




- minimize $\|J - I\|_2^2 + \sum_{1\le i,j\le n} \min\{|J(i,j)|,\ |1-J(i,j)|\}$

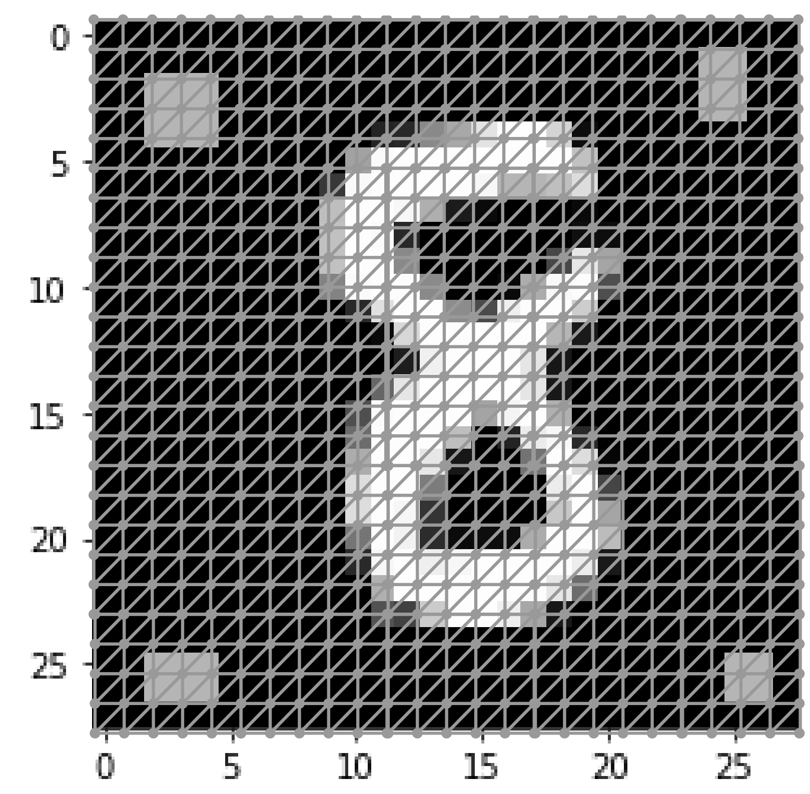
# Example: image binarization [Carrière et al. '21]

**Input:** greyscaled image $I \colon \{1, \cdots, n\}^2 \to [0, 1]$.

**Output:** image $J : \{1, \cdots, n\}^2 \to \{0, 1\}$

- $X$ = grid $\{1, \cdots, n\}^2$ triangulated
- $F(I)$ = upper-star filtration of $I$

$$\left| \begin{array}{l} F(I)(v) = I(v) \\ F(I)(\{u, v\}) = \min\{I(u), I(v)\} \end{array} \right.$$



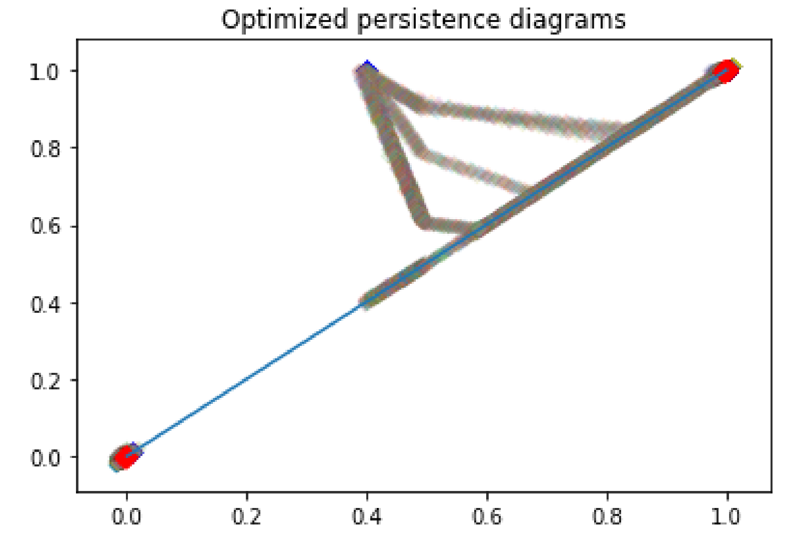

$X$

- minimize $\|J - I\|_2^2 + \sum_{1 \le i,j \le n} \min\{|J(i,j)|, |1 - J(i,j)|\}$

# Example: image binarization [Carrière et al. '21]

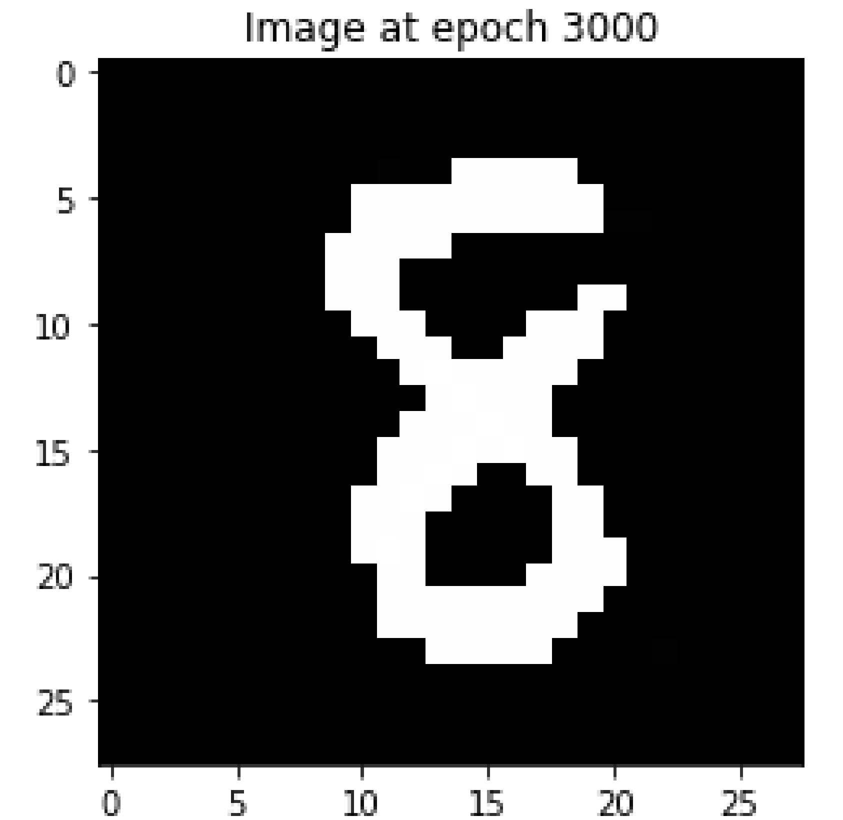
**Input:** greyscaled image $I\colon \{1, \cdots, n\}^2 \to [0,1]$.

**Output:** image $J : \{1, \cdots, n\}^2 \to \{0,1\}$




- $X$ = grid $\{1, \cdots, n\}^2$ triangulated
- $F(I)$ = upper-star filtration of $I$

$$\left| \begin{array}{l} F(I)(v) = I(v) \\ F(I)(\{u,v\}) = \min\{I(u), I(v)\} \end{array} \right.$$




- $\mathcal{L} \circ V(D) = \displaystyle\sum_{(x,y)\in D} (y-x)^2$
- minimize $\|J - I\|_2^2 + \displaystyle\sum_{1 \le i,j \le n} \min\{|J(i,j)|, |1 - J(i,j)|\} + \mathcal{L} \circ V \circ \mathrm{Pers}_0 \circ F$

# Example: image binarization [Carrière et al. '21]

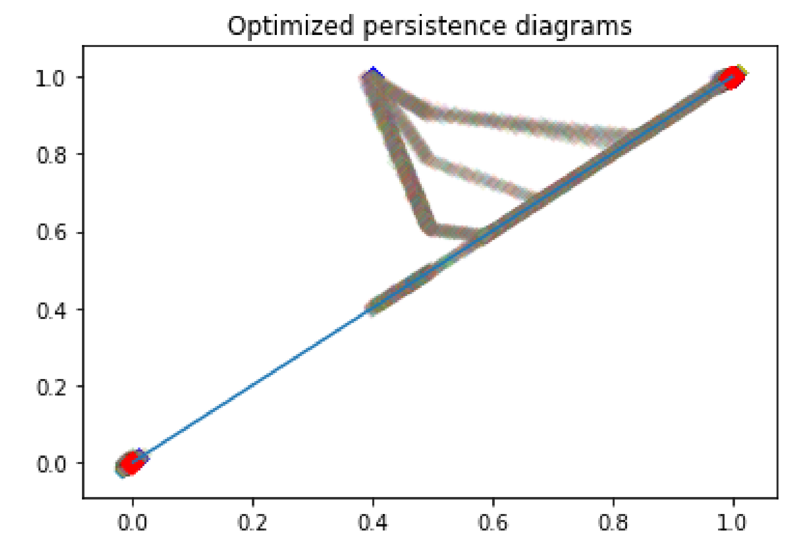
**Input:** greyscaled image $I \colon \{1, \cdots, n\}^2 \to [0, 1]$.

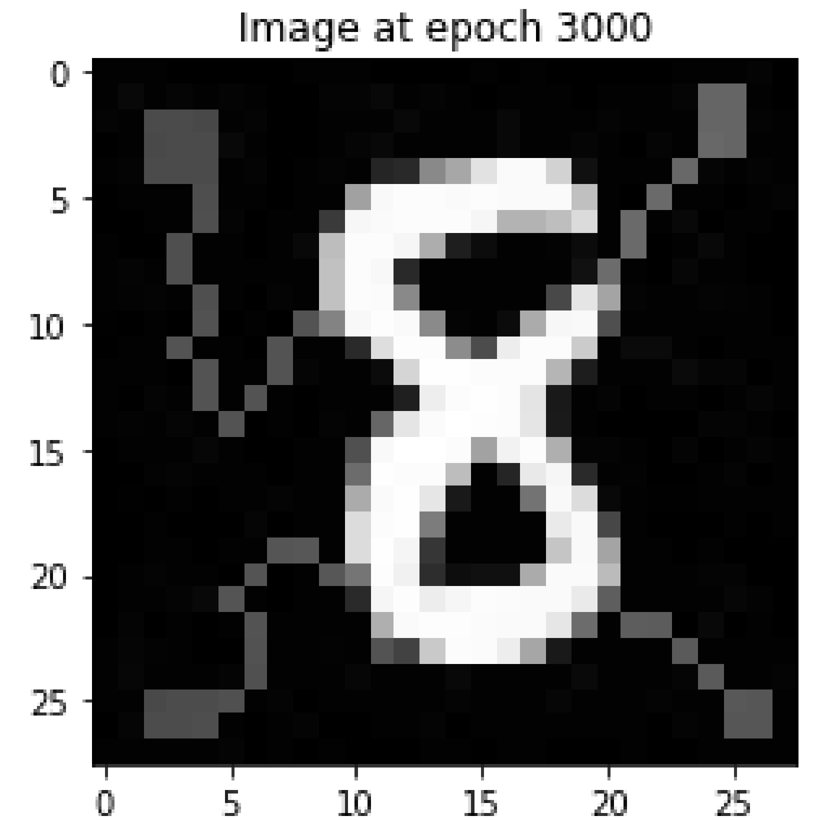
**Output:** image $J : \{1, \cdots, n\}^2 \to \{0, 1\}$

- $X$ = grid $\{1, \cdots, n\}^2$ triangulated
- $F(I)$ = upper-star filtration of $I$

$$\left| \begin{array}{l} F(I)(v) = I(v) \\ F(I)(\{u, v\}) = \min\{I(u), I(v)\} \end{array} \right.$$

- $\mathcal{L} \circ V(D) = \sum_{(x,y) \in D} (y - x)^2$
- minimize $\|J - I\|_2^2 + \mathcal{L} \circ V \circ \mathrm{Pers}_0 \circ F$

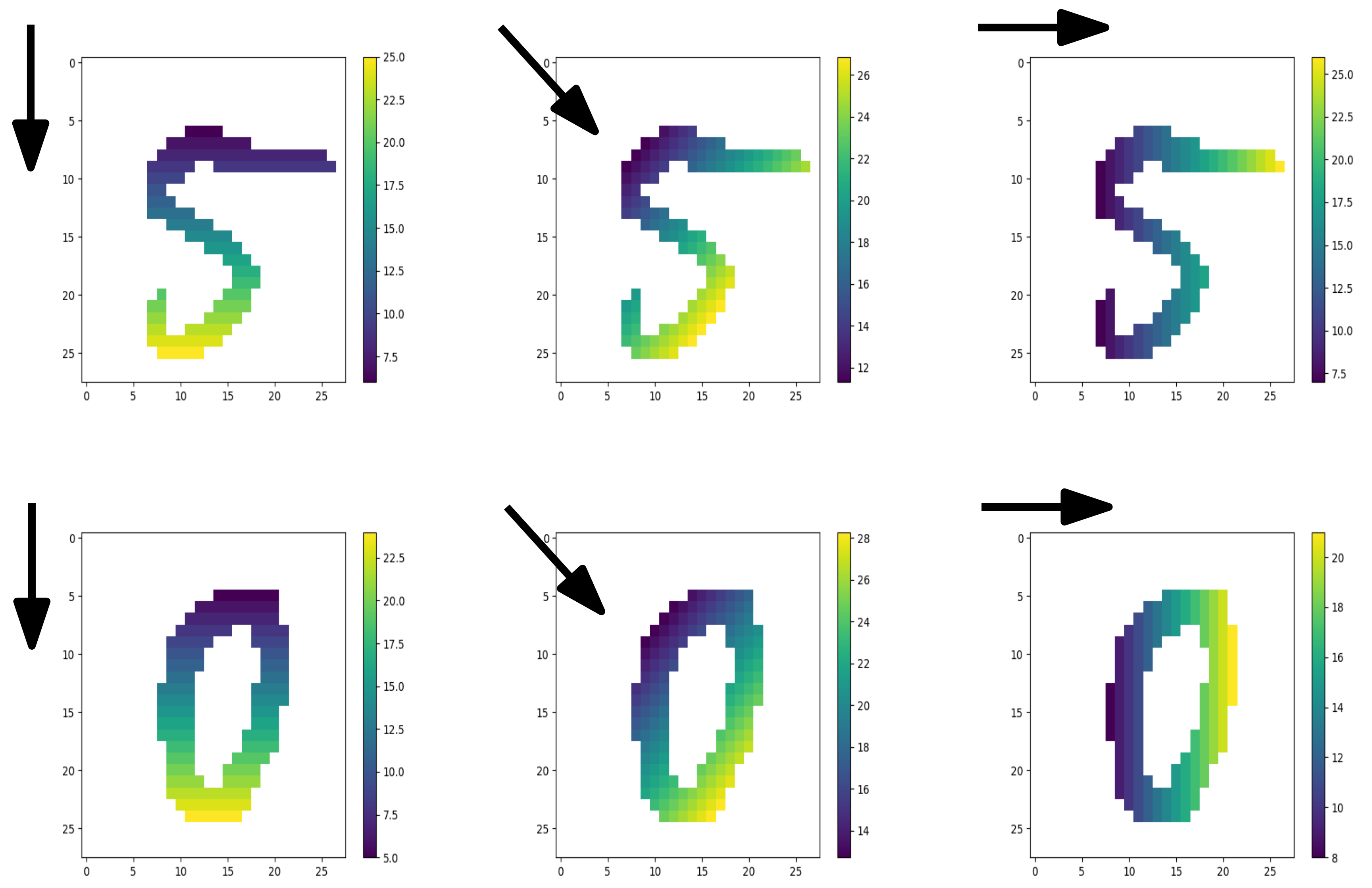


# Example: orientation selection [Carrière et al. ’21]

**Input:** MNIST dataset

**Goal:** given two classes $0 \leq i \neq j \leq 9$, optimize orientation $\theta_{i,j}$ so that RF performs best at distinguishing between the two classes from the barcodes of the projections along $\theta_{i,j}$.



# Example: orientation selection [Carrière et al. '21]

**Input:** MNIST dataset

**Goal:** given two classes $0 \leq i \neq j \leq 9$, optimize orientation $\theta_{i,j}$ so that RF performs best at distinguishing between the two classes from the barcodes of the projections along $\theta_{i,j}$.

**Results:**

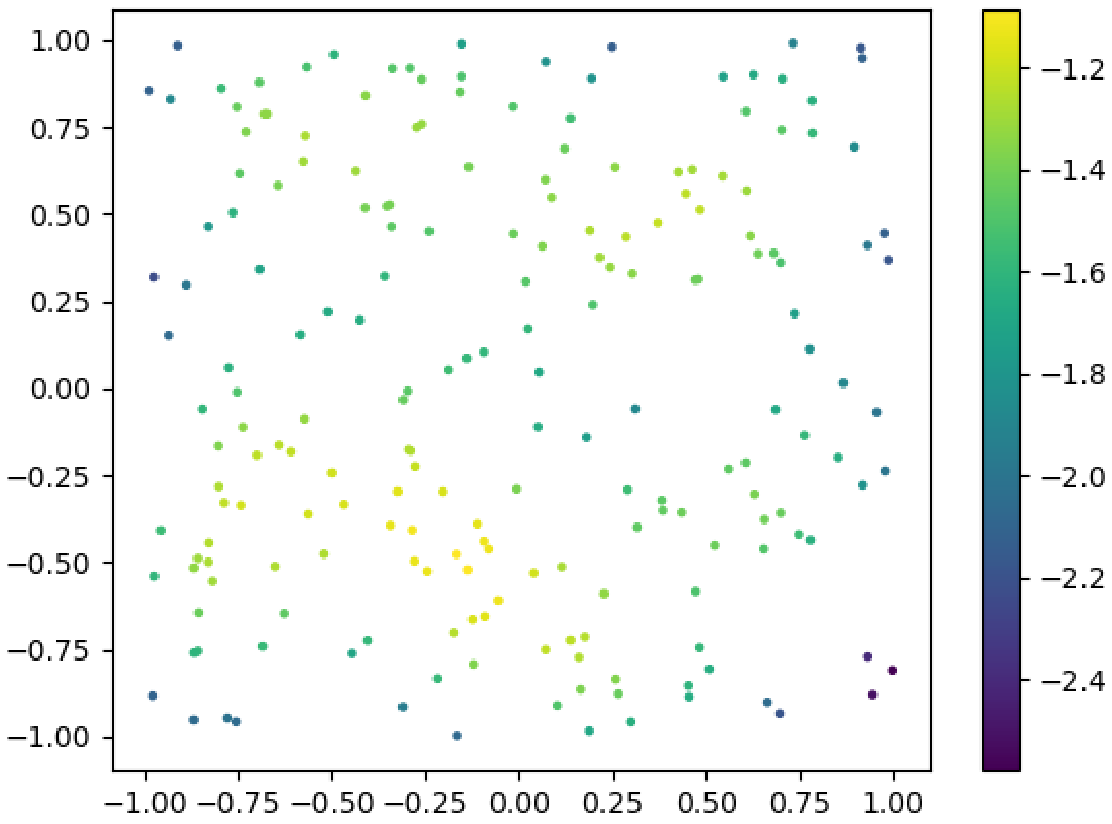
| Dataset | Baseline | Before | After | Difference | Dataset | Baseline | Before | After | Difference |
|---|---|---|---|---|---|---|---|---|---|
| vs01 | 100.0 | 61.3 | 99.0 | **+37.6** | vs26 | 99.7 | 98.8 | 98.2 | -0.6 |
| vs02 | 99.4 | 98.8 | 97.2 | -1.6 | vs28 | 99.1 | 96.8 | 96.8 | 0.0 |
| vs06 | 99.4 | 87.3 | 98.2 | **+10.9** | vs29 | 99.1 | 91.6 | 98.6 | **+7.0** |
| vs09 | 99.4 | 86.8 | 98.3 | **+11.5** | vs34 | 99.8 | 99.4 | 99.1 | -0.3 |
| vs16 | 99.7 | 89.0 | 97.3 | **+8.3** | vs36 | 99.7 | 99.3 | 99.3 | -0.1 |
| vs19 | 99.6 | 84.8 | 98.0 | **+13.2** | vs37 | 98.9 | 94.9 | 97.5 | **+2.6** |
| vs24 | 99.4 | 98.7 | 98.7 | 0.0 | vs57 | 99.7 | 90.5 | 97.2 | **+6.7** |
| vs25 | 99.4 | 80.6 | 97.2 | **+16.6** | vs79 | 99.1 | 85.3 | 96.9 | **+11.5** |

vs$ij$: class $i$ vs. class $j$

baseline: RF applied to raw images

# Example: point cloud optimization

$$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2,\ a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$$



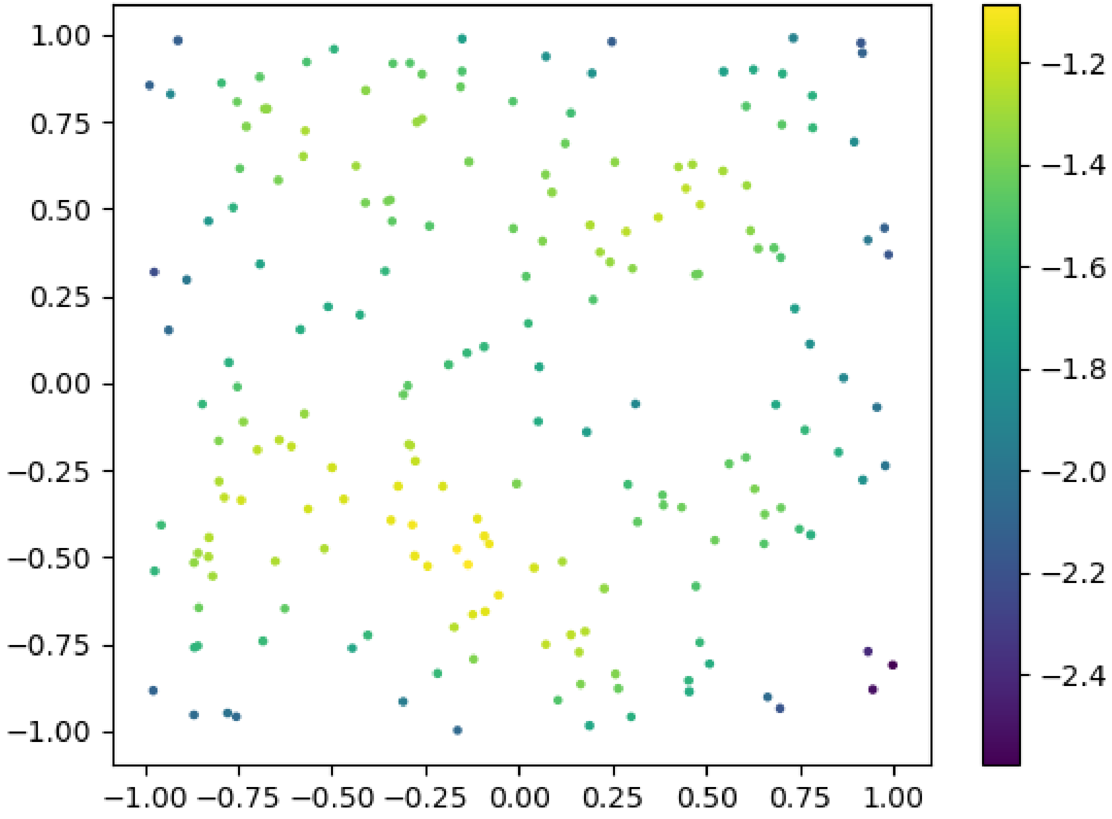
# Example: point cloud optimization

$$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2,\ a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$$

$$X = 2^{[\![1,r]\!]},\ m = 2r$$

$$F(A)\colon \left| \begin{array}{l} 2^{[\![1,r]\!]} \to \mathbb{R} \\ \sigma \mapsto \operatorname{diam}\ \sigma \end{array} \right.$$ (Rips filtration)

$$\mathcal{L} \circ V\,(D) := \sum_{(x,y)\in D} x - y$$



# Example: point cloud optimization

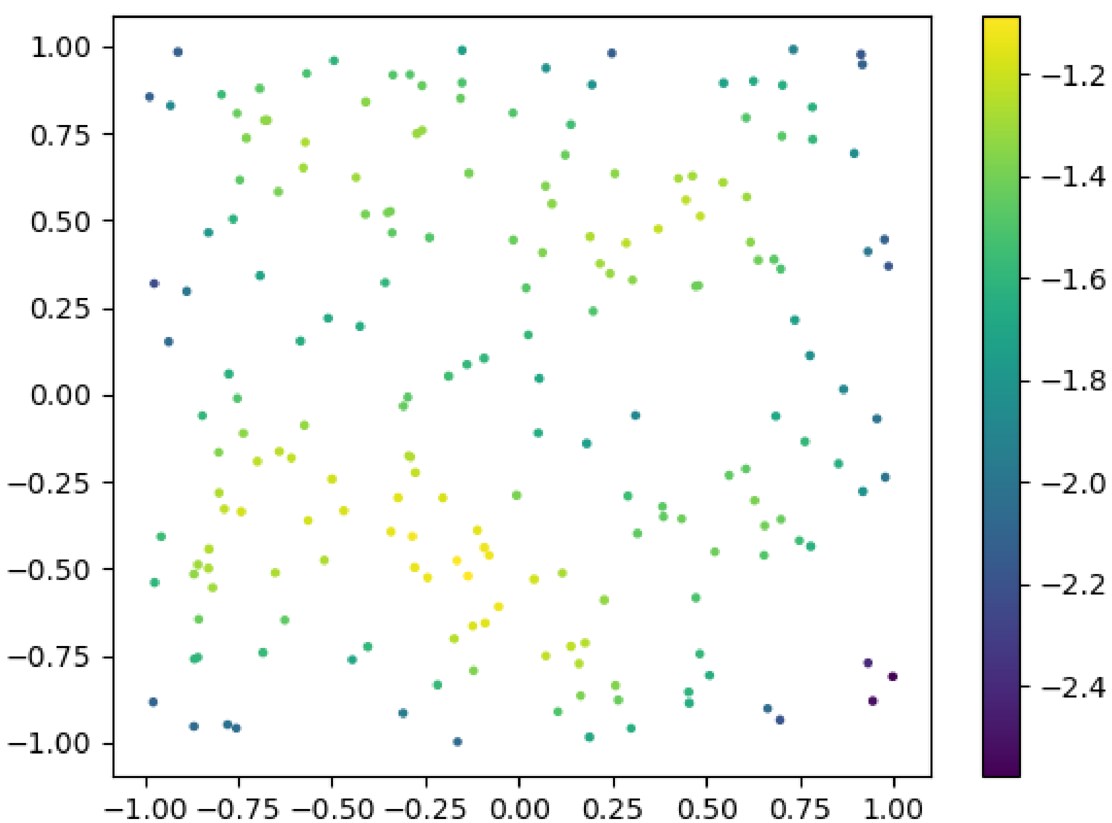
$$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2,\ a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$$

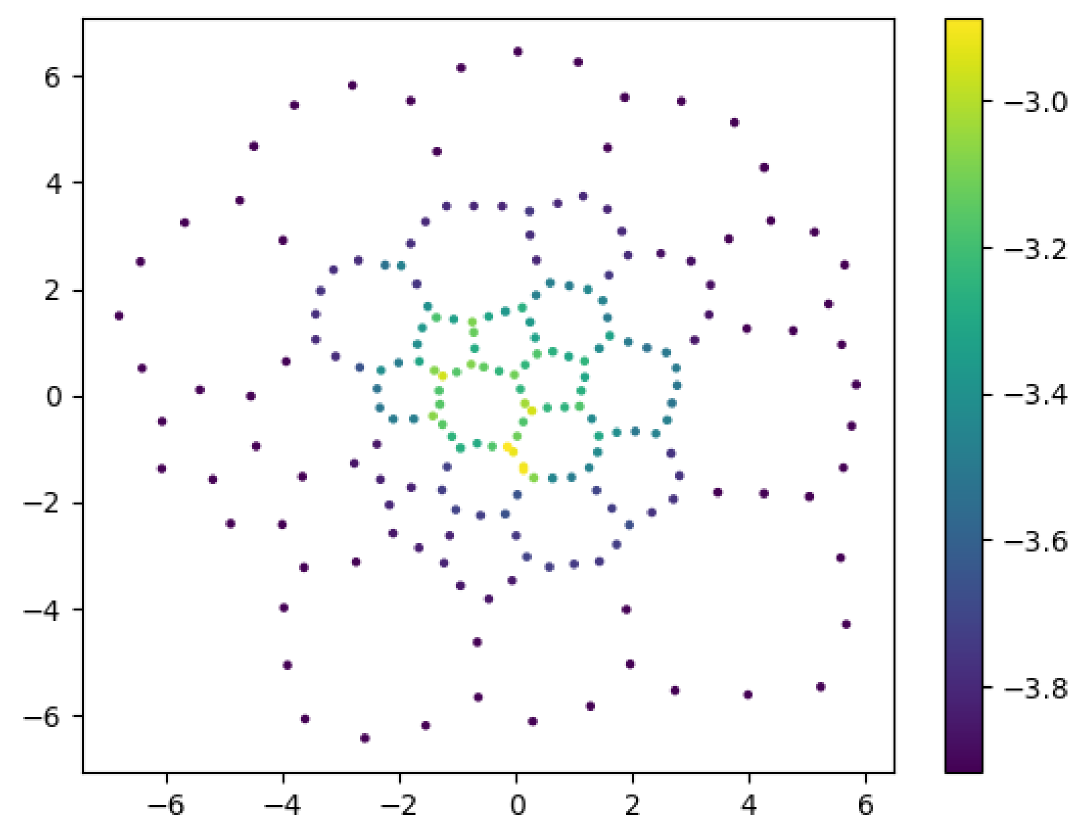
$$X = 2^{[\![1,r]\!]},\ m = 2r$$

$$F(A)\colon \left| \begin{array}{l} 2^{[\![1,r]\!]} \to \mathbb{R} \\ \sigma \mapsto \operatorname{diam}\ \sigma \end{array} \right.$$ (Rips filtration)

Loss$(A) = \mathcal{L} \circ V \circ \mathrm{Pers}_1 \circ F(A)$

$$\mathcal{L} \circ V\,(D) := \sum_{(x,y)\in D} x - y$$




epoch: 100

# Example: point cloud optimization

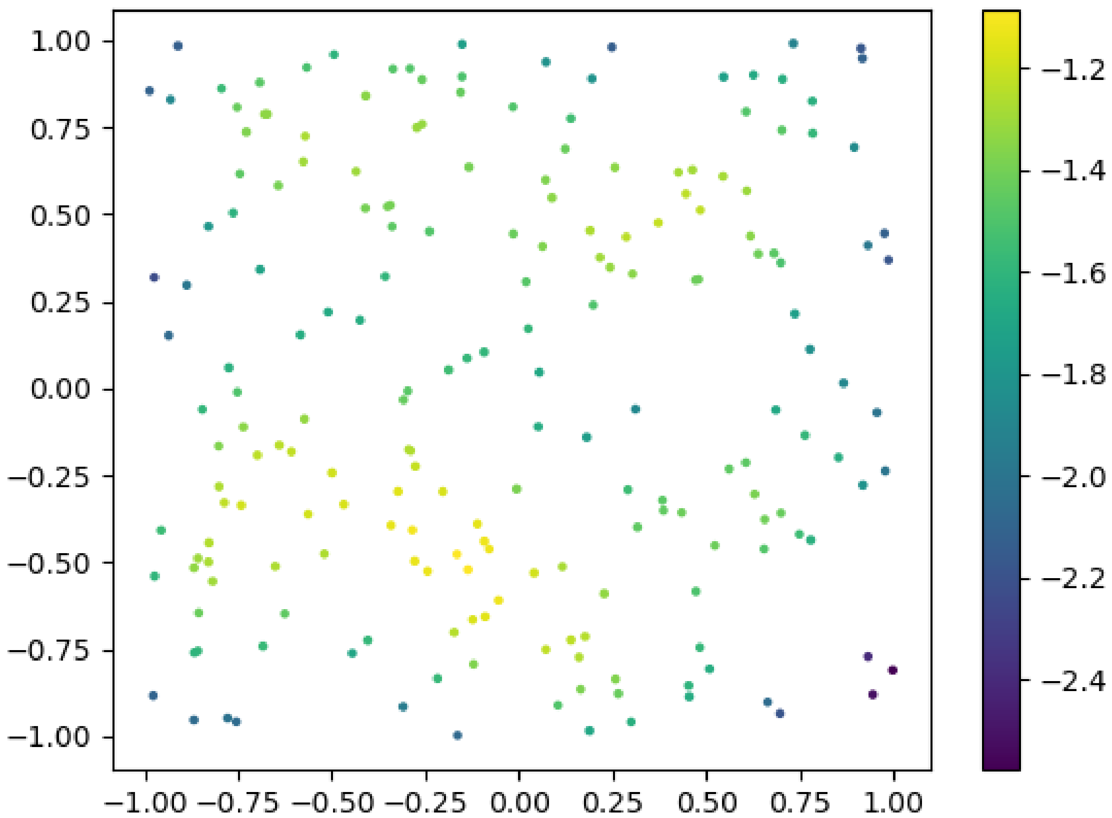
$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2$, $a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$

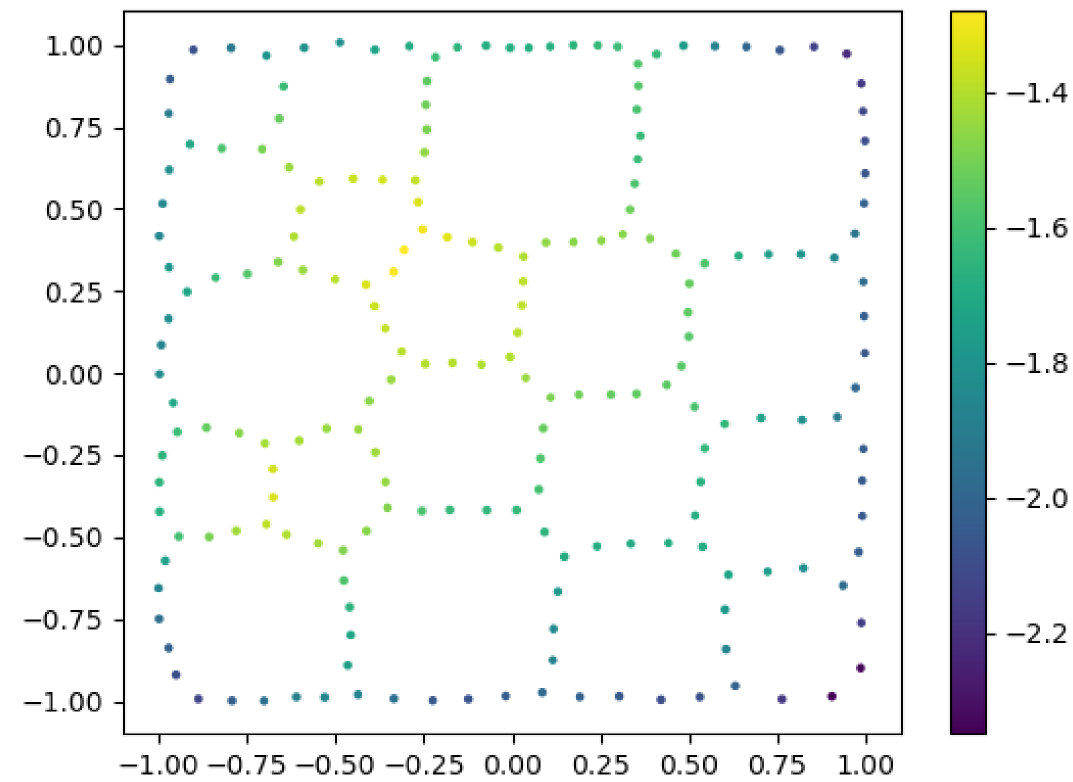
$X = 2^{[\![1,r]\!]}$, $m = 2r$

$$F(A) \colon \left| \begin{array}{l} 2^{[\![1,r]\!]} \to \mathbb{R} \\ \sigma \mapsto \operatorname{diam}\ \sigma \end{array} \right.$$ (Rips filtration)

$$\text{Loss}(A) = \mathcal{L} \circ V \circ \mathrm{Pers}_1 \circ F(A)$$
$$+ \lambda \textstyle\sum_i \max\{0,\ \|a_i\|_1 - 1\}$$

$$\mathcal{L} \circ V\,(D) := \sum_{(x,y) \in D} x - y$$




epoch: 100

# Example: point cloud optimization

$$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2,\ a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$$

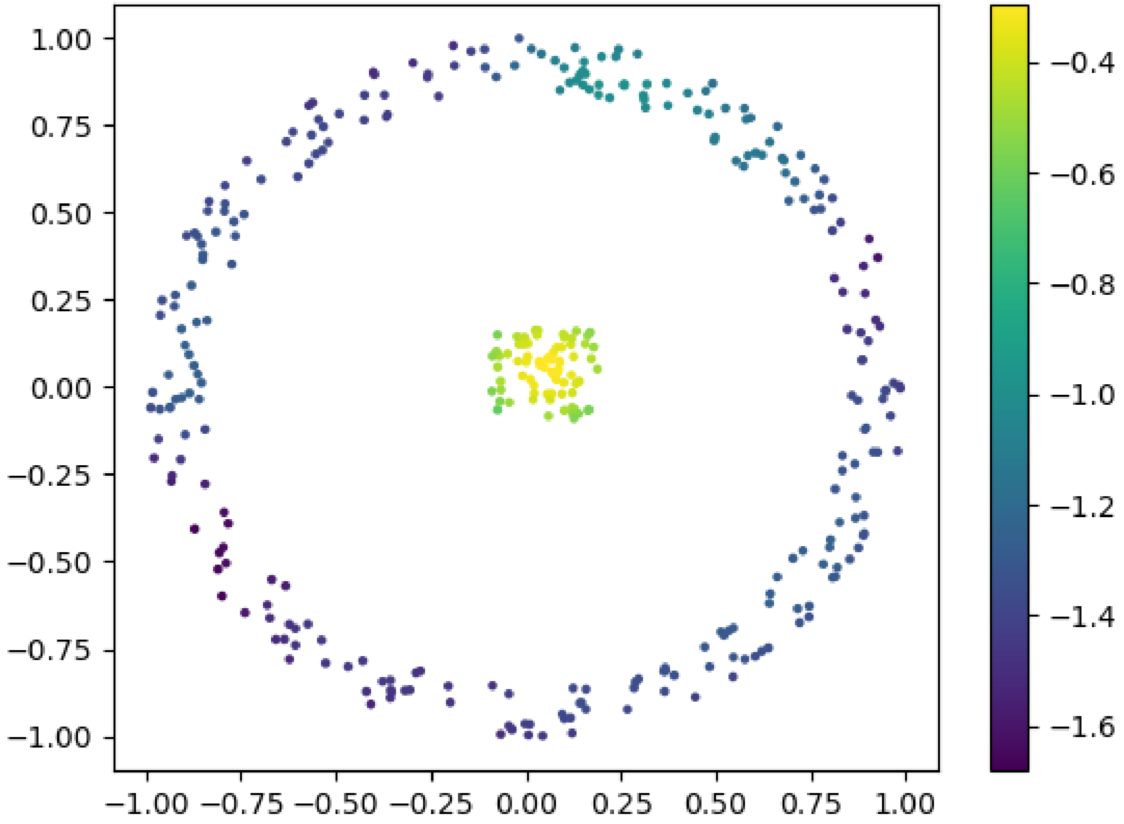
$$X = 2^{[\![1,r]\!]},\ m = 2r$$

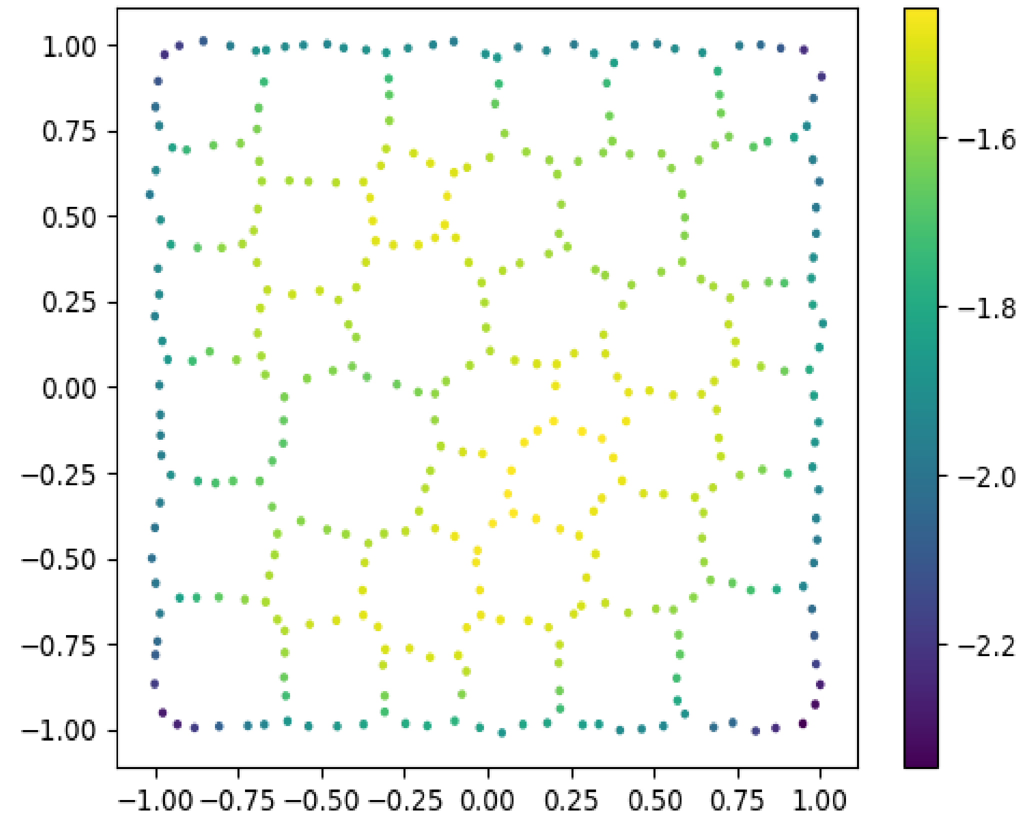
$$F(A)\colon \left| \begin{array}{l} 2^{[\![1,r]\!]} \to \mathbb{R} \\ \sigma \mapsto \mathrm{diam}\ \ \sigma \end{array} \right.$$ (Rips filtration)

$$\mathcal{L} \circ V\,(D) := \sum_{(x,y)\in D} x - y$$

Loss$(A) = \mathcal{L} \circ V \circ \mathrm{Pers}_1 \circ F(A)$

$$+\lambda \textstyle\sum_i \max\{0,\ \|a_i\|_1 - 1\}$$




epoch: 100

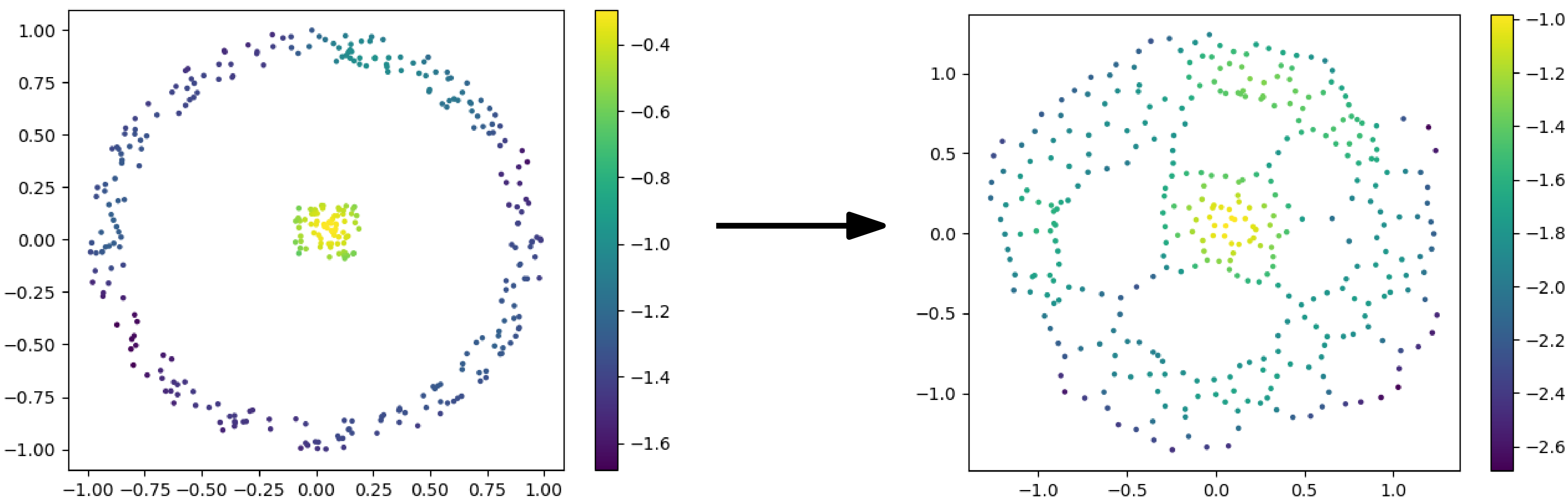
# Example: point cloud optimization

$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2$, $a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$

$X = 2^{[\![1,r]\!]}$, $m = 2r$

$F(A) := (\mathsf{Rips}(A), \text{-kde}_A) : 2^{[\![1,r]\!]} \to \mathbb{R}^2$

$\mathsf{Loss}(A) = \mathcal{L} \circ V \circ \mathrm{Pers}_1 \circ F(A)$




epoch: 100

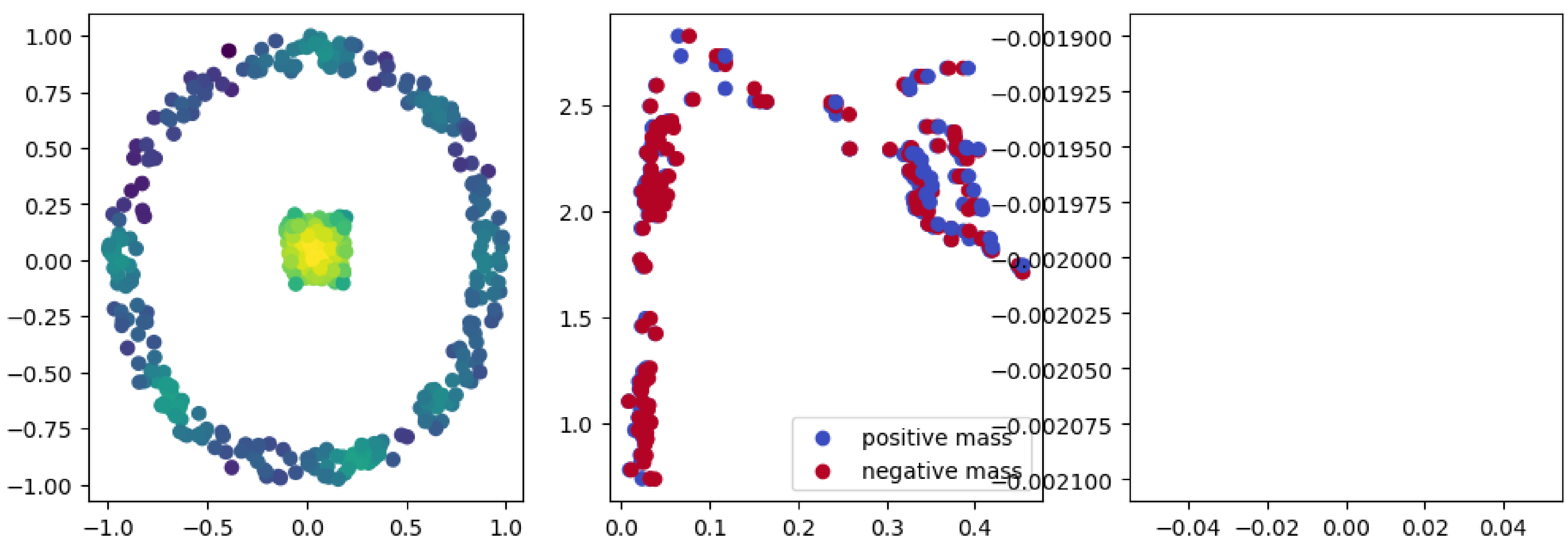
# Example: point cloud optimization

$$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2,\ a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$$

$$X = 2^{[\![1,r]\!]},\ m = 2r$$

$$F(A) := (\mathsf{Rips}(A),\ \text{-kde}_A) : 2^{[\![1,r]\!]} \to \mathbb{R}^2$$

$$\mathsf{Loss}(A) = \mathcal{L} \circ V \circ \mathrm{Pers}_1 \circ F(A)$$



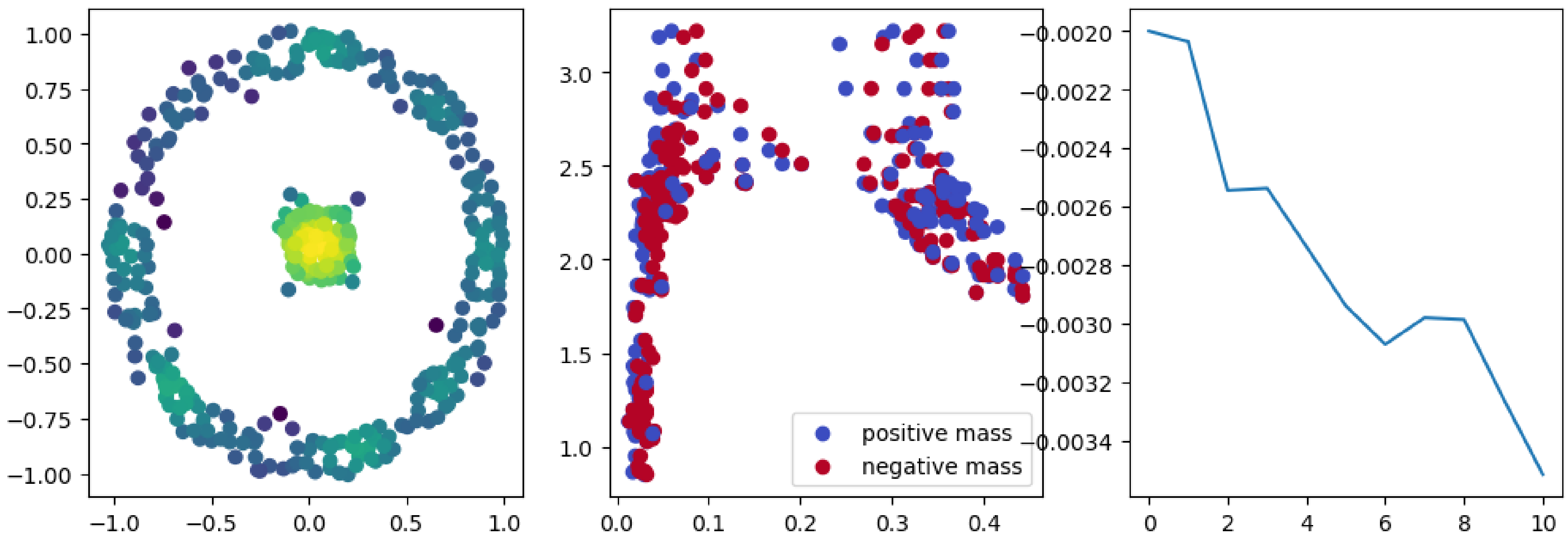
# Example: point cloud optimization

$$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2,\ a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$$

$$X = 2^{[\![1,r]\!]},\ m = 2r$$

$$F(A) := (\mathsf{Rips}(A),\ \text{-kde}_A) : 2^{[\![1,r]\!]} \to \mathbb{R}^2$$

$$\mathsf{Loss}(A) = \mathcal{L} \circ V \circ \mathrm{Pers}_1 \circ F(A)$$



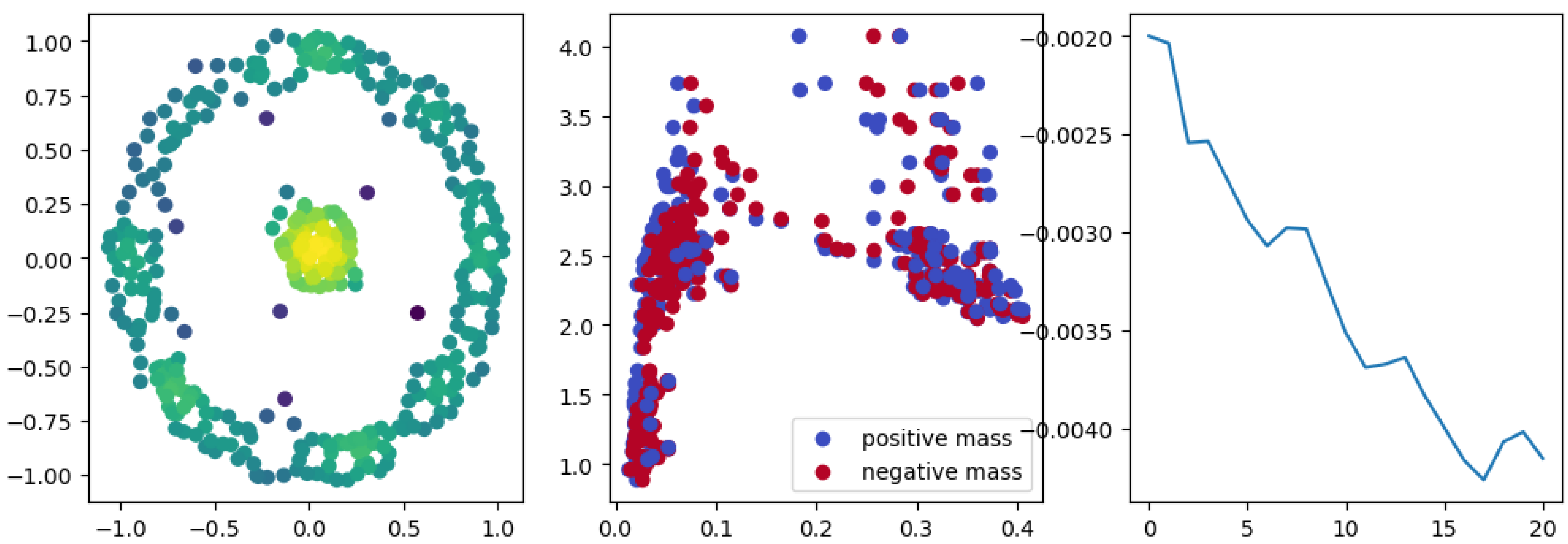
# Example: point cloud optimization

$$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2,\ a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$$

$$X = 2^{[\![1,r]\!]},\ m = 2r$$

$$F(A) := (\mathsf{Rips}(A),\ \text{-kde}_A) : 2^{[\![1,r]\!]} \to \mathbb{R}^2$$

$$\mathsf{Loss}(A) = \mathcal{L} \circ V \circ \mathrm{Pers}_1 \circ F(A)$$



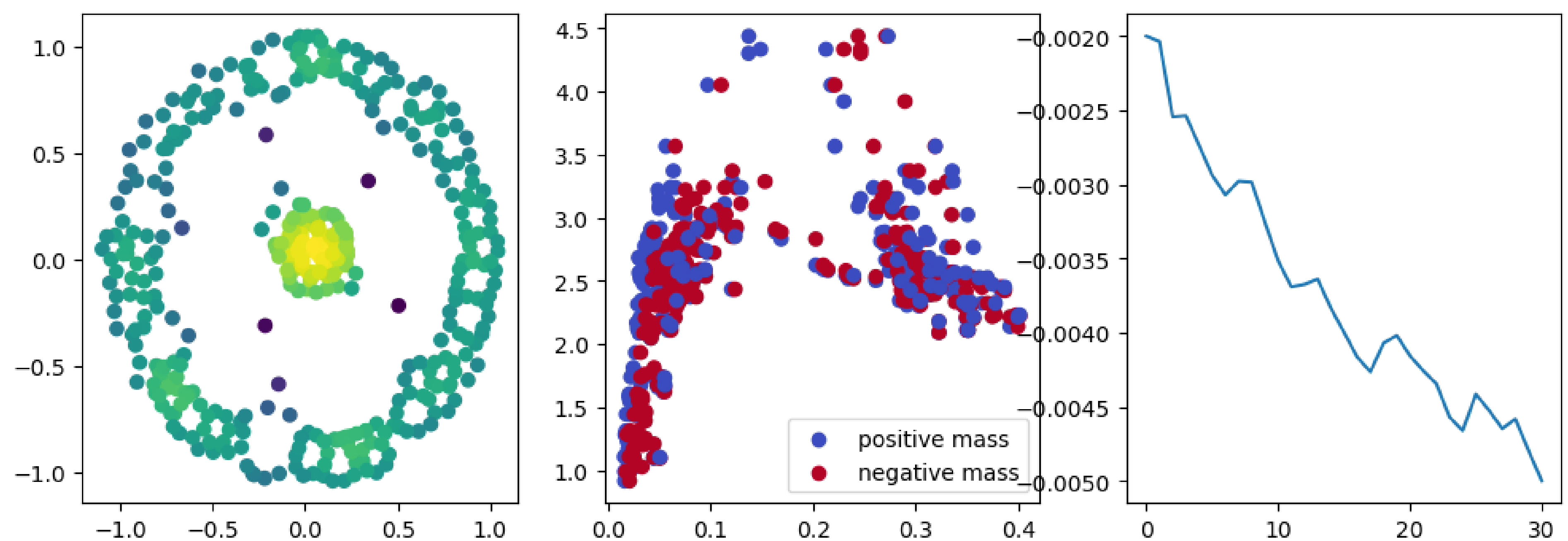
# Example: point cloud optimization

$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2$, $a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$

$X = 2^{[\![1,r]\!]}$, $m = 2r$

$F(A) := (\mathsf{Rips}(A), \text{-kde}_A) : 2^{[\![1,r]\!]} \to \mathbb{R}^2$

$\mathsf{Loss}(A) = \mathcal{L} \circ V \circ \mathrm{Pers}_1 \circ F(A)$



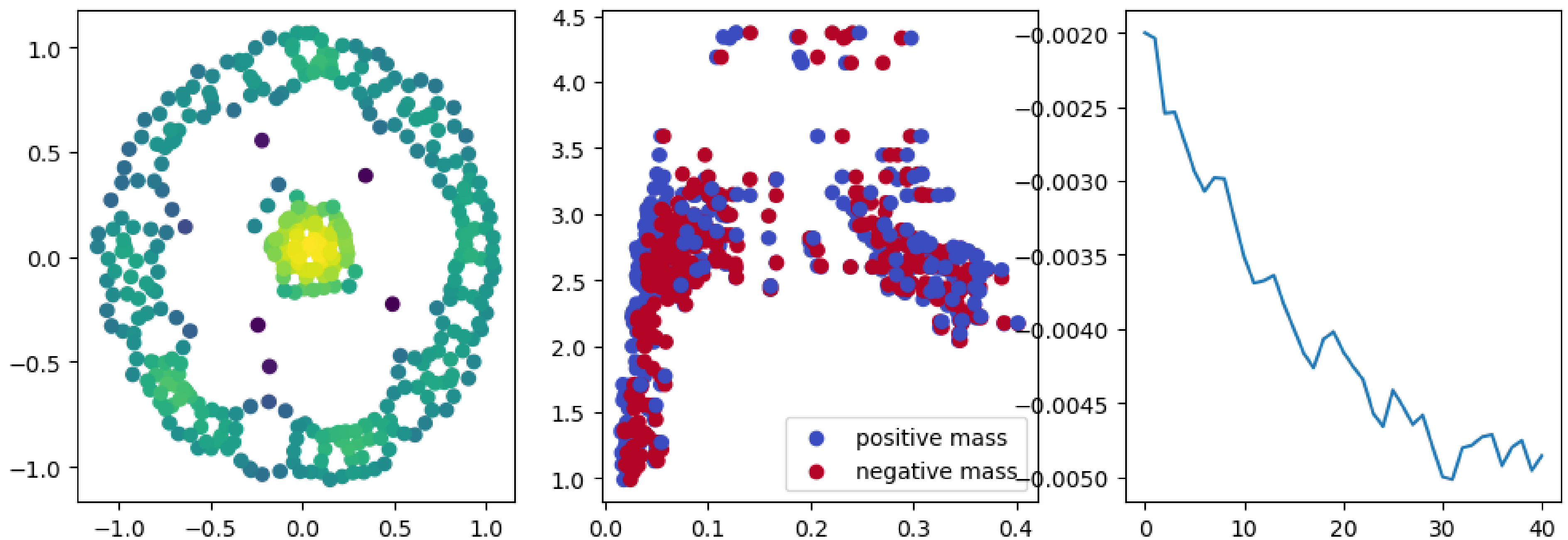
# Example: point cloud optimization

$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2$, $a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$

$X = 2^{[\![1,r]\!]}$, $m = 2r$

$F(A) := (\mathsf{Rips}(A), \text{-kde}_A) : 2^{[\![1,r]\!]} \to \mathbb{R}^2$

$$\mathsf{Loss}(A) = \mathcal{L} \circ V \circ \mathrm{Pers}_1 \circ F(A)$$



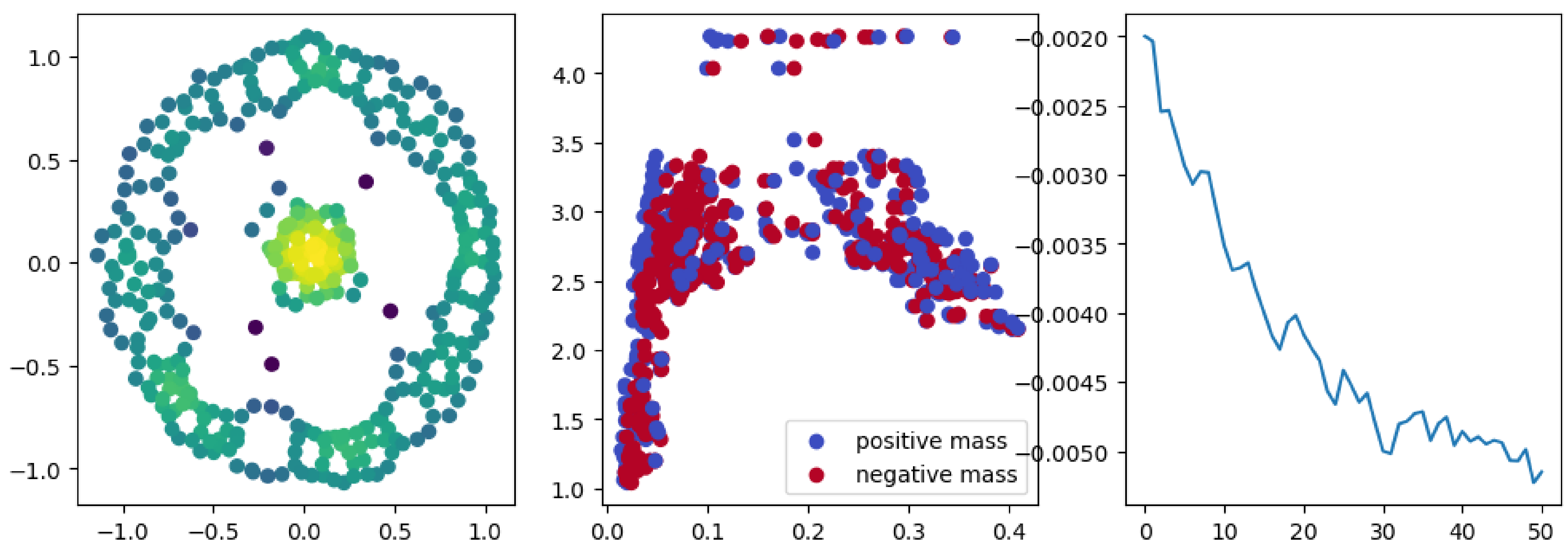
# Example: point cloud optimization

$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2$, $a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$

$X = 2^{[\![1,r]\!]}$, $m = 2r$

$F(A) := (\mathsf{Rips}(A), \text{-kde}_A) : 2^{[\![1,r]\!]} \to \mathbb{R}^2$

$\mathsf{Loss}(A) = \mathcal{L} \circ V \circ \mathrm{Pers}_1 \circ F(A)$



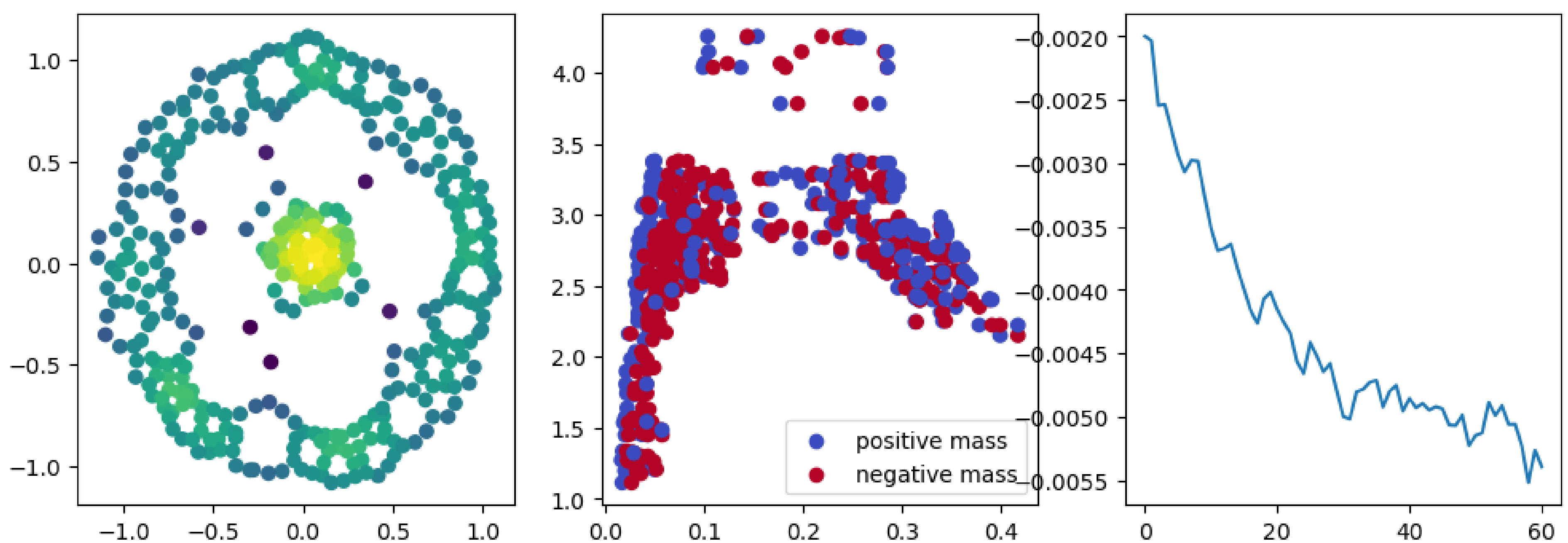
# Example: point cloud optimization

$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2$, $a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$

$X = 2^{[\![1,r]\!]}$, $m = 2r$

$F(A) := (\mathsf{Rips}(A), \text{-kde}_A) : 2^{[\![1,r]\!]} \to \mathbb{R}^2$

$$\mathsf{Loss}(A) = \mathcal{L} \circ V \circ \mathrm{Pers}_1 \circ F(A)$$



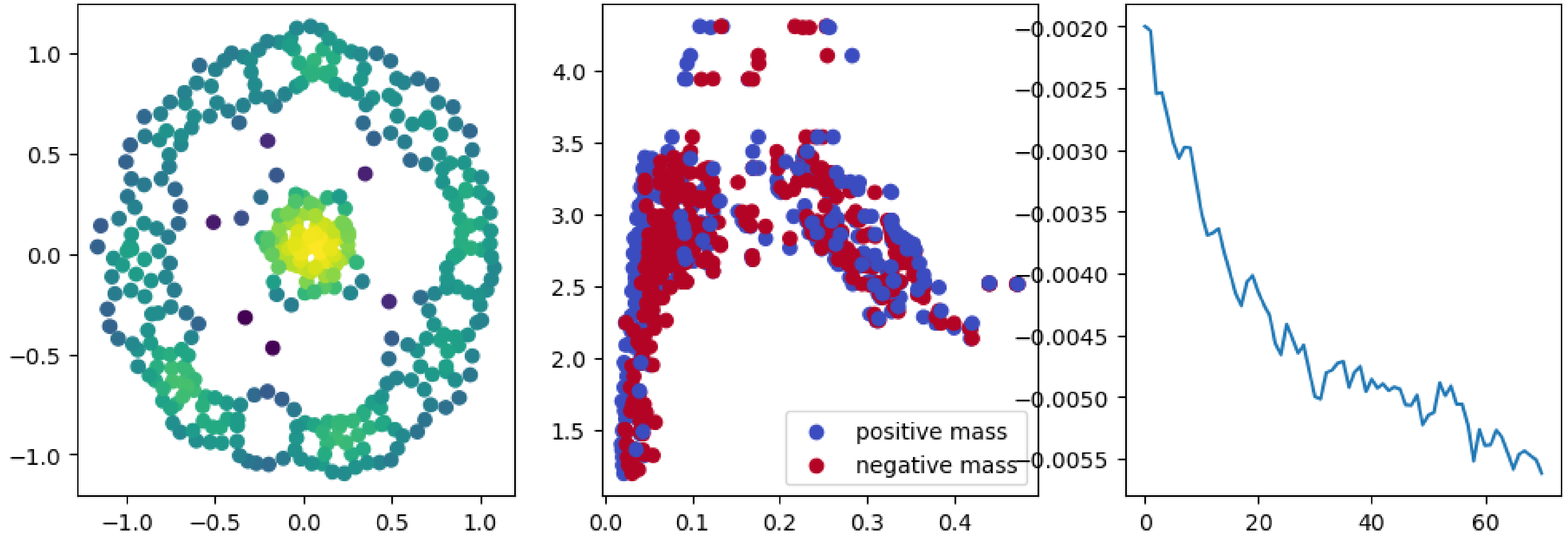
# Example: point cloud optimization

$$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2,\ a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$$

$$X = 2^{[\![1,r]\!]},\ m = 2r$$

$$F(A) := (\mathsf{Rips}(A),\ \text{-kde}_A) : 2^{[\![1,r]\!]} \to \mathbb{R}^2$$

$$\mathsf{Loss}(A) = \mathcal{L} \circ V \circ \mathrm{Pers}_1 \circ F(A)$$



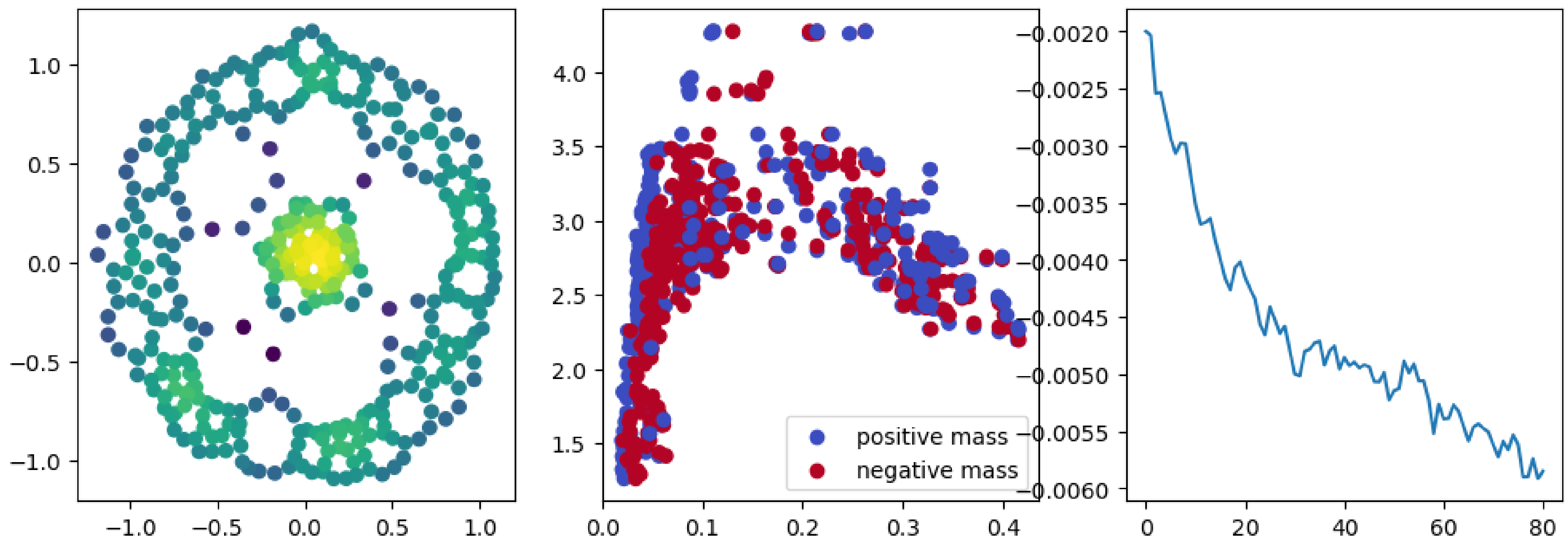
# Example: point cloud optimization

$$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2,\ a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$$

$$X = 2^{[\![1,r]\!]},\ m = 2r$$

$$F(A) := (\mathsf{Rips}(A),\ \text{-kde}_A) : 2^{[\![1,r]\!]} \to \mathbb{R}^2$$

$$\mathsf{Loss}(A) = \mathcal{L} \circ V \circ \mathrm{Pers}_1 \circ F(A)$$



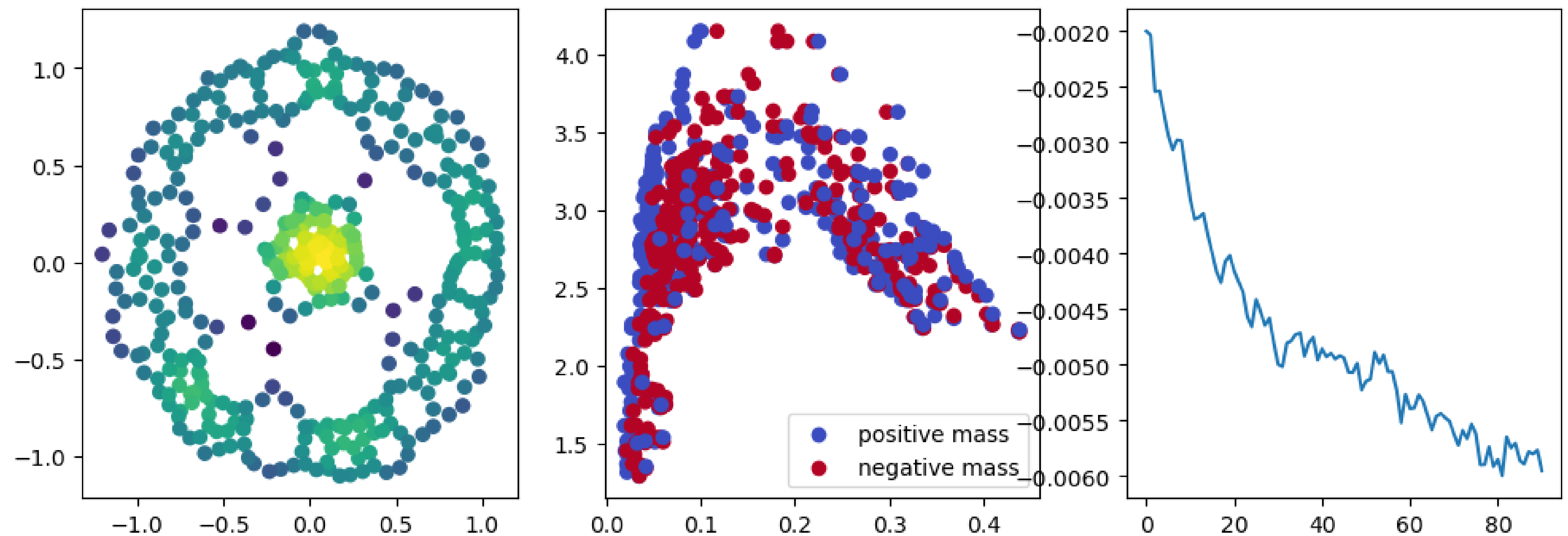
# Example: point cloud optimization

$A = \{a_1, \cdots, a_r\} \subset \mathbb{R}^2$, $a_i \overset{\text{iid}}{\sim} \mathcal{U}\left([-1,1]^2\right)$

$X = 2^{[\![1,r]\!]}$, $m = 2r$

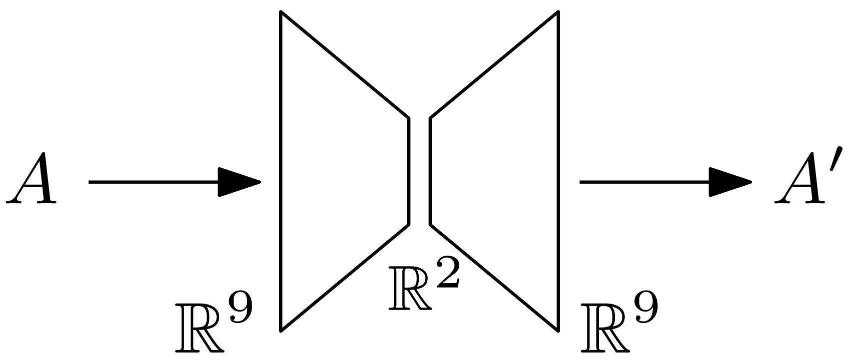
$F(A) := (\mathsf{Rips}(A), \text{-kde}_A) : 2^{[\![1,r]\!]} \to \mathbb{R}^2$

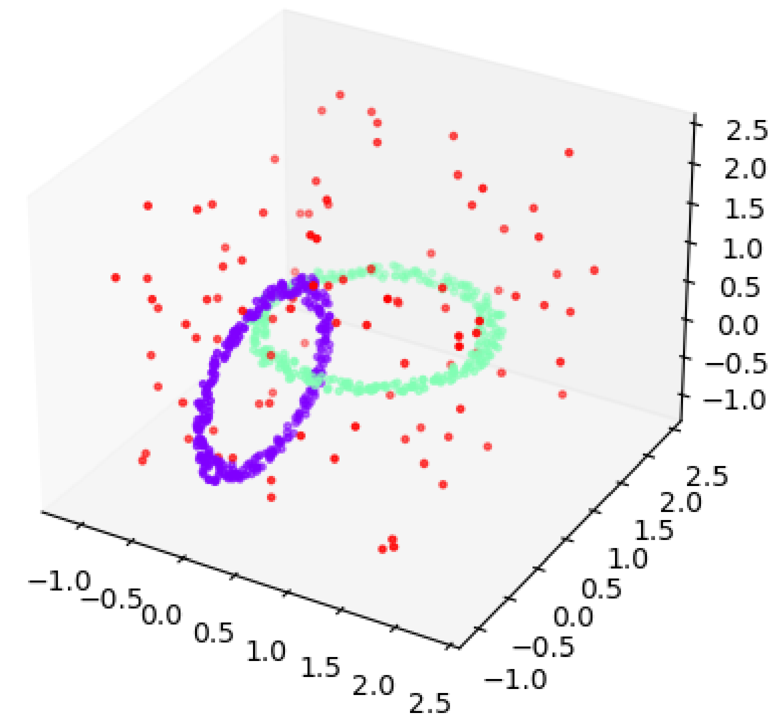
$\mathsf{Loss}(A) = \mathcal{L} \circ V \circ \mathrm{Pers}_1 \circ F(A)$

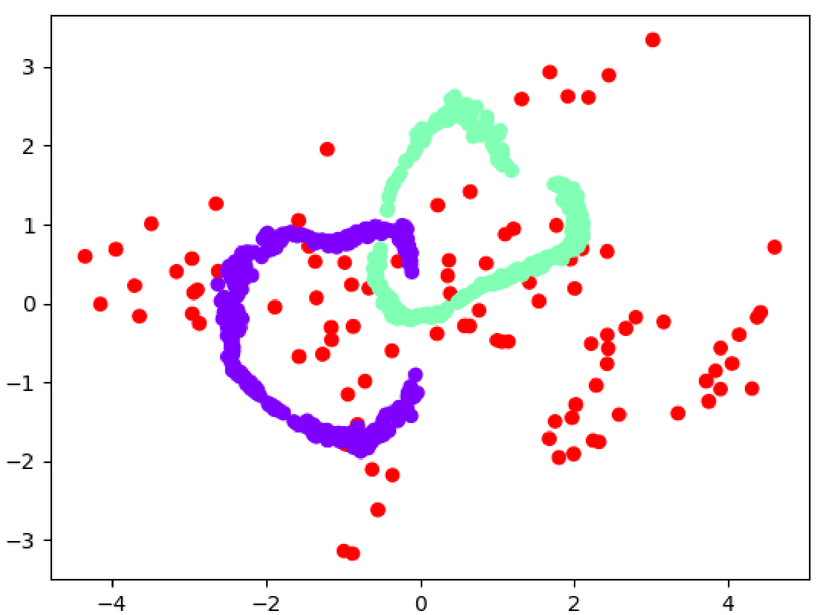


# Example: dimensionality reduction

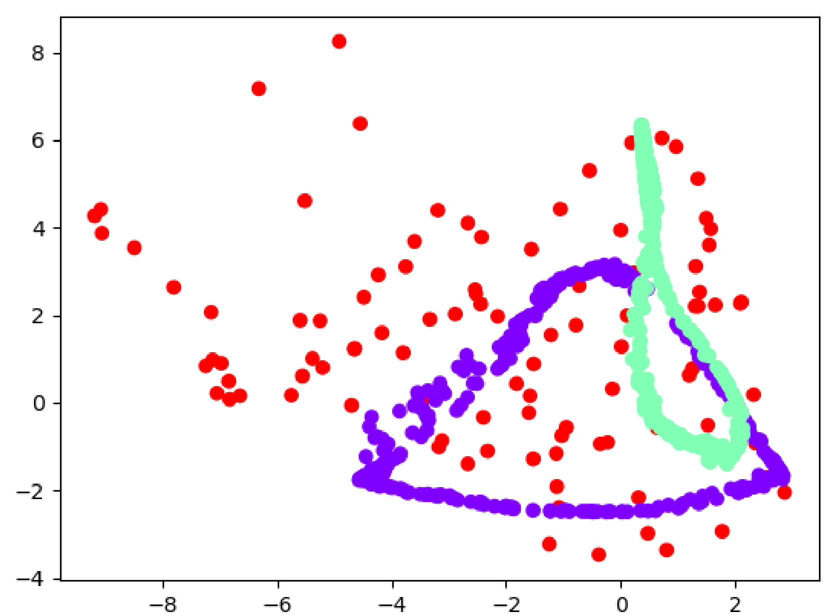
$A$ sampled from 2 interlaced circles + clutter in $\mathbb{R}^9$



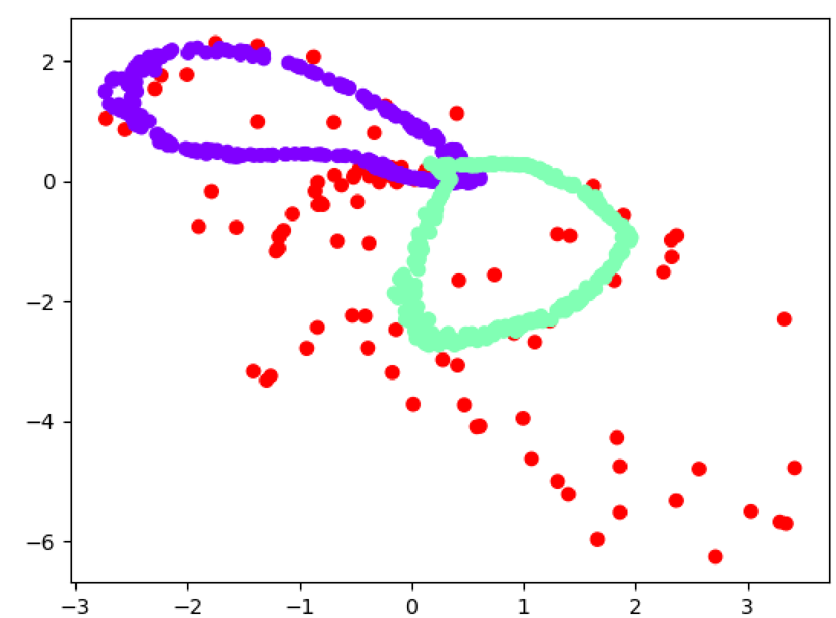





autoencoder



autoencoder + topo. reg. w/ $\mathrm{d}_A$



autoencoder + topo. reg. w/ $(\mathrm{d}_A, \text{-}\mathrm{kde}_A)$

# Example: graph classification

GNN produces $n$ attributes for nodes or edges in the input graph

Topological layer inserted after GNN layer produces barcodes

Barcodes are combined with attributes for an enriched graph representation

Representation is learnt in an **end-to-end** fashion in graph classification tasks

| Model | ENZYMES | IMDB-B | IMDB-M | MUTAG |
|---|---|---|---|---|
| GCN | 30.3±8.1 | 73.2±6.4 | 44.9±7.6 | **87.2±5.6** |
| GCN+1pTOP | 28.8±7.5 | 75.2±5.6 | **51.2±4.4** | 84.1±8.9 |
| GCN+npTOP | **39.0±10.1** | **78.4±5.1** | 51.1±3.5 | 85.1±7.7 |
| GIN | **47.0±12.9** | 71.2±5.4 | 47.1±2.9 | 87.2±8.0 |
| GIN+1pTOP | 45.3±11.8 | **75.0±2.7** | 47.5±5.0 | **88.3±8.9** |
| GIN+npTOP | 46.5±11.2 | 71.3±5.1 | **48.5±4.2** | 87.2±6.1 |
| GraphResNet | 42.8±11.0 | **75.3±5.3** | 49.4±4.3 | 88.8±5.2 |
| GraphResNet+1pTOP | 39.5±12.2 | 68.1±8.2 | 40.7±3.5 | 87.8±4.3 |
| GraphResNet+npTOP | **44.3±9.8** | 69.4±5.8 | **50.1±4.4** | **89.3±6.1** |
| GraphDenseNet | 43.2±10.4 | 50.3±5.9 | 33.1±2.7 | 88.8±5.2 |
| GraphDenseNet+1pTOP | 47.3±12.3 | 50.0±7.1 | 32.7±4.2 | 86.2±8.3 |
| GraphDenseNet+npTOP | **48.0±11.4** | **52.2±7.7** | **34.1±3.1** | **92.6±5.1** |