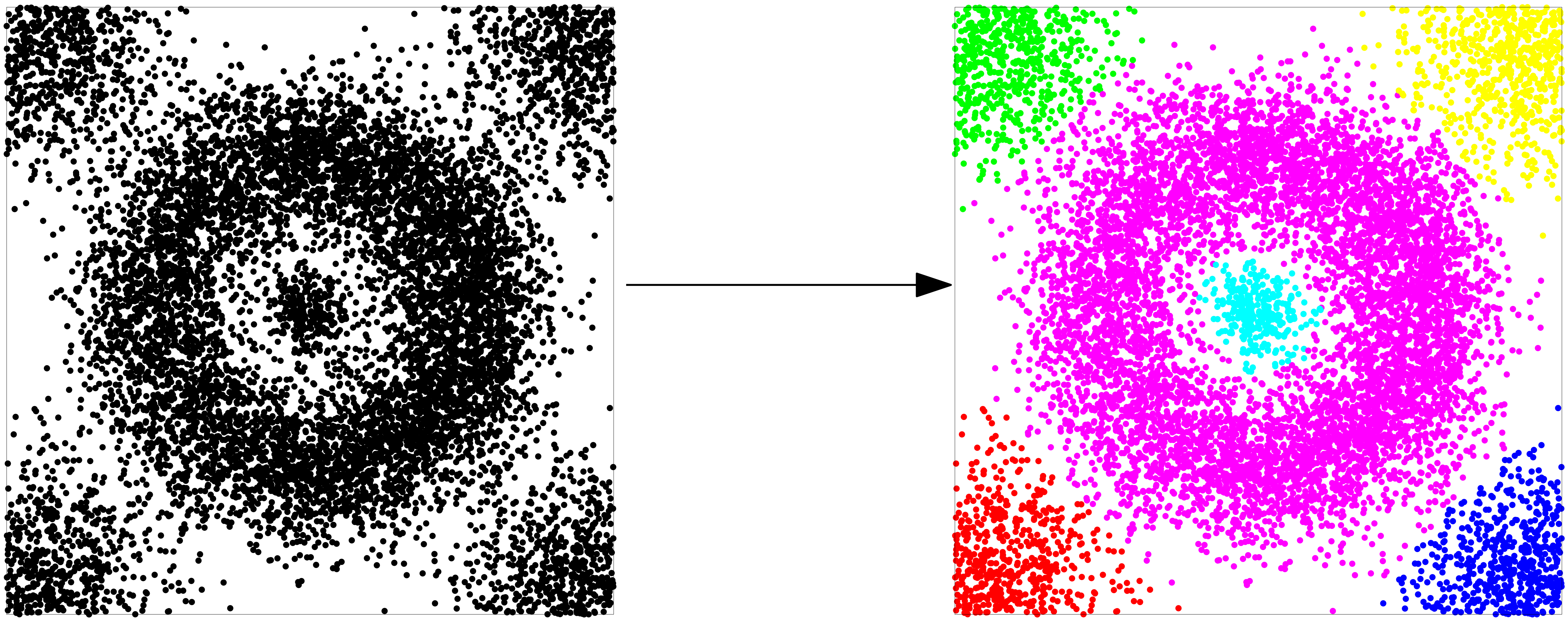
Clustering

Cluster Analysis

Input: a finite set of observations: - point cloud with coordinates

- distance / (dis-)similarity matrix



Task:

partition the data points into a collection of *relevant* subsets called clusters

A Wealth of Approaches

Variational

- k-means / k-medoid
- EM
- CLARA

spectral k-means

Spectral

- Normalized Cut
- Multiway Cut

Hierarchical divisive/agglomerative

- single-linkage
- BIRCH

Density thresholding

- DBSCAN
- OPTICS

Mode seeking

- Mean/Medoid/Quick Shift
- graph-based hill climbing

Valley seeking

- [JBD'79]
- NDDs [ZZZL'07]

A Wealth of Approaches

Variational

- k-means / k-medoid
- EM
- CLARA

spectral k-means

Spectral

- Normalized Cut
- Multiway Cut

Hierarchical divisive/agglomerative

- single-linkage
- BIRCH

Density thresholding

- DBSCAN
- OPTICS

Mode seeking

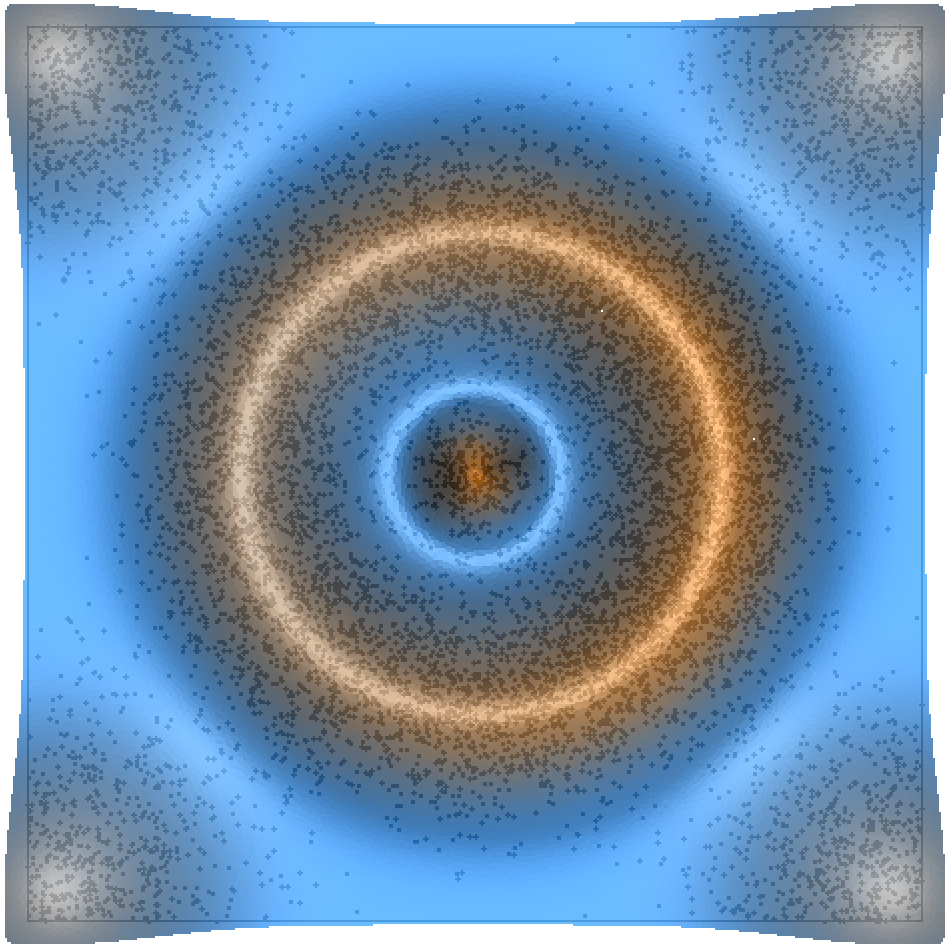
- Mean/Medoid/Quick Shift
- graph-based hill climbing

Valley seeking

- [JBD'79]
- NDDs [ZZZL'07]

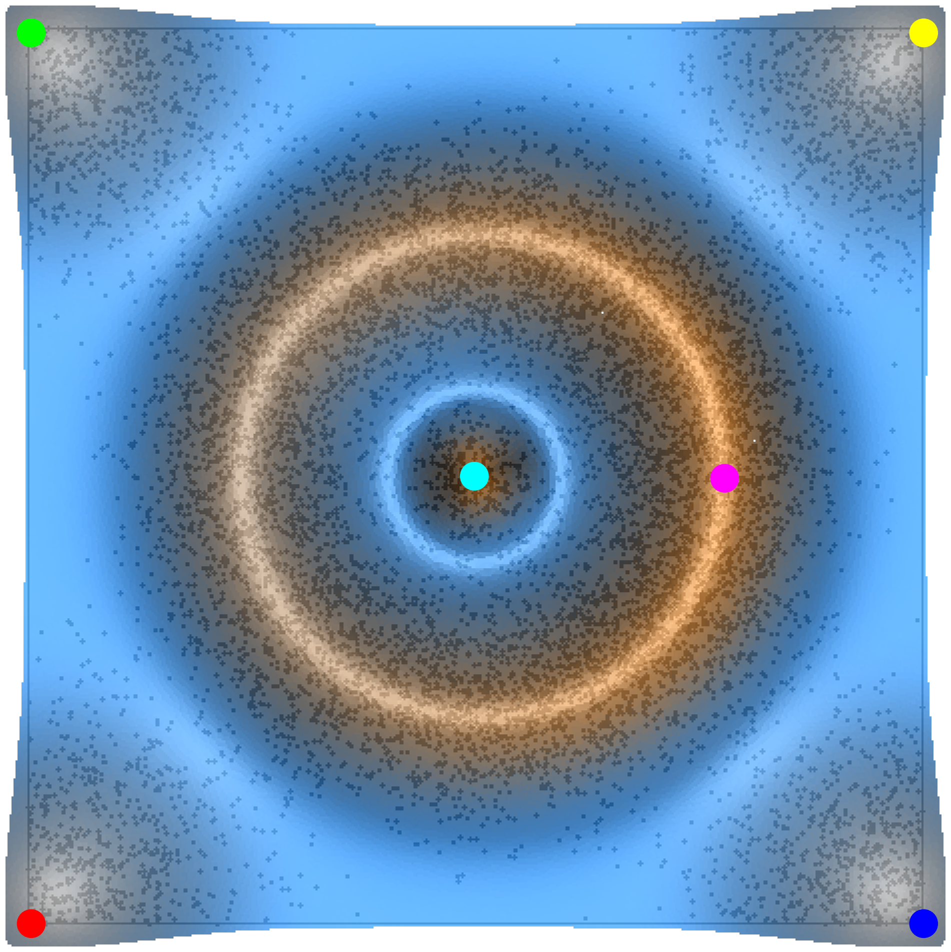
Mode-Seeking Paradigm

- Assume the data points are sampled from some unknown probability distribution
- Partition the data according to the basins of attraction of the peaks of the density



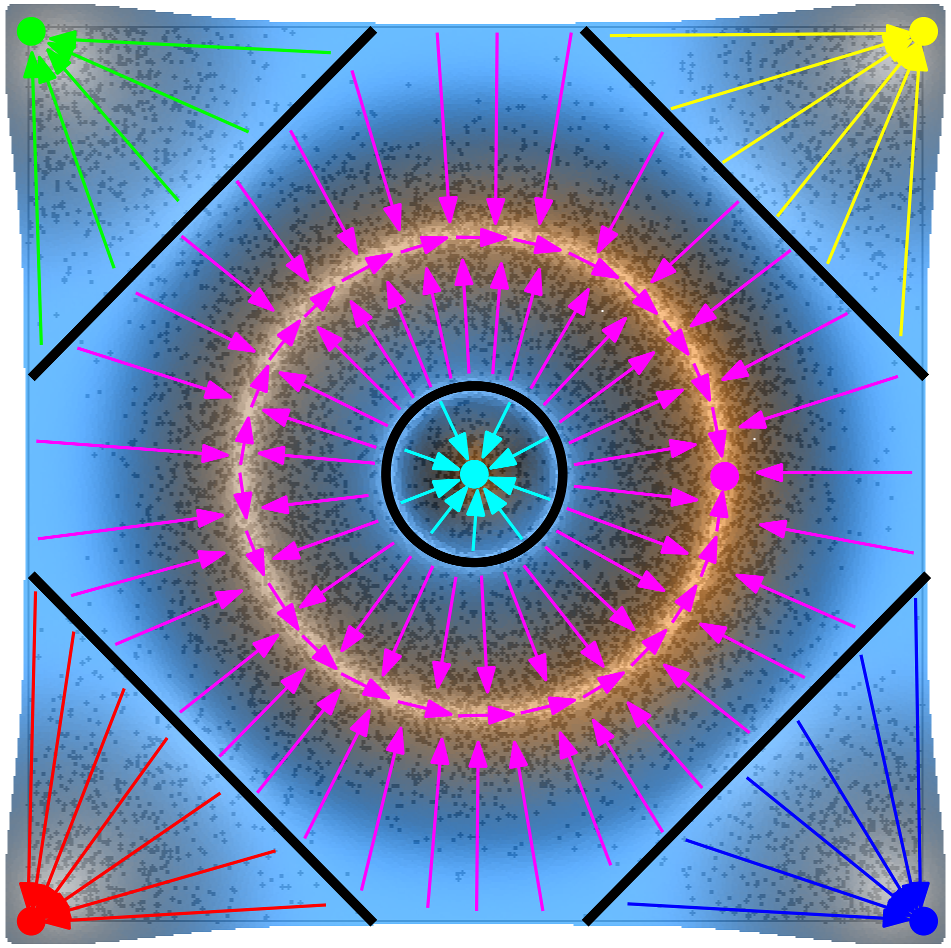
Mode-Seeking Paradigm

- Assume the data points are sampled from some unknown probability distribution
- Partition the data according to the basins of attraction of the peaks of the density



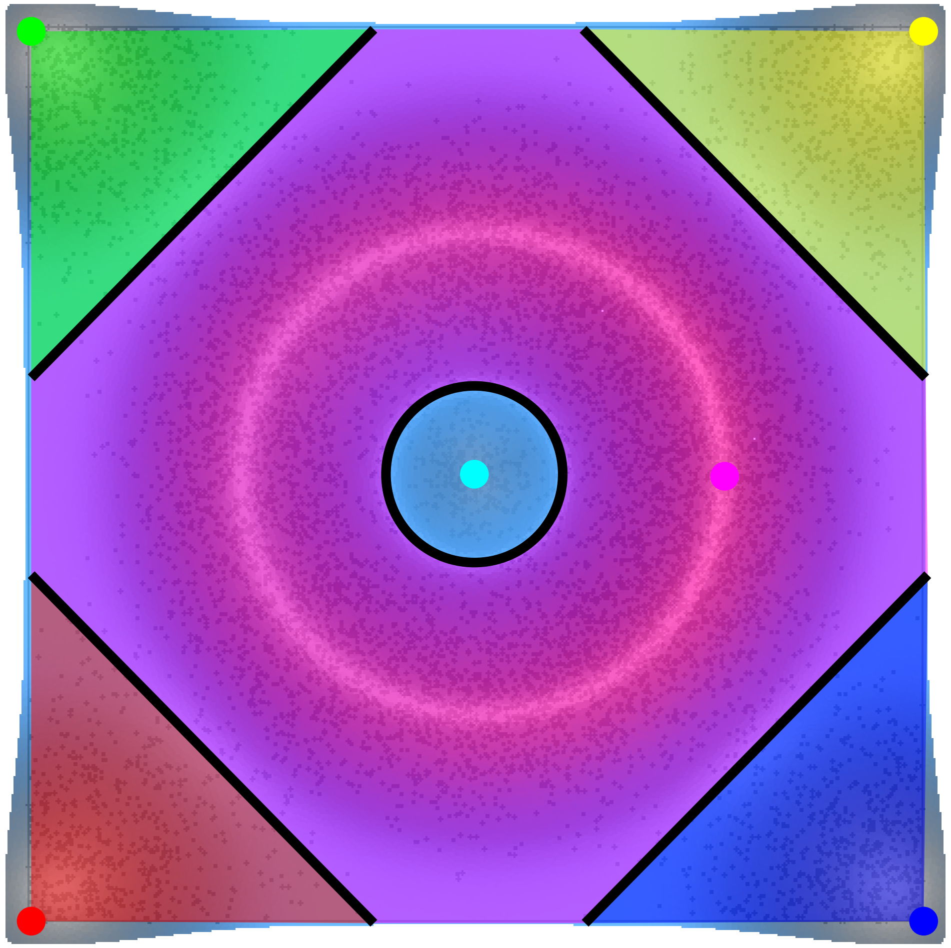
Mode-Seeking Paradigm

- Assume the data points are sampled from some unknown probability distribution
- Partition the data according to the basins of attraction of the peaks of the density



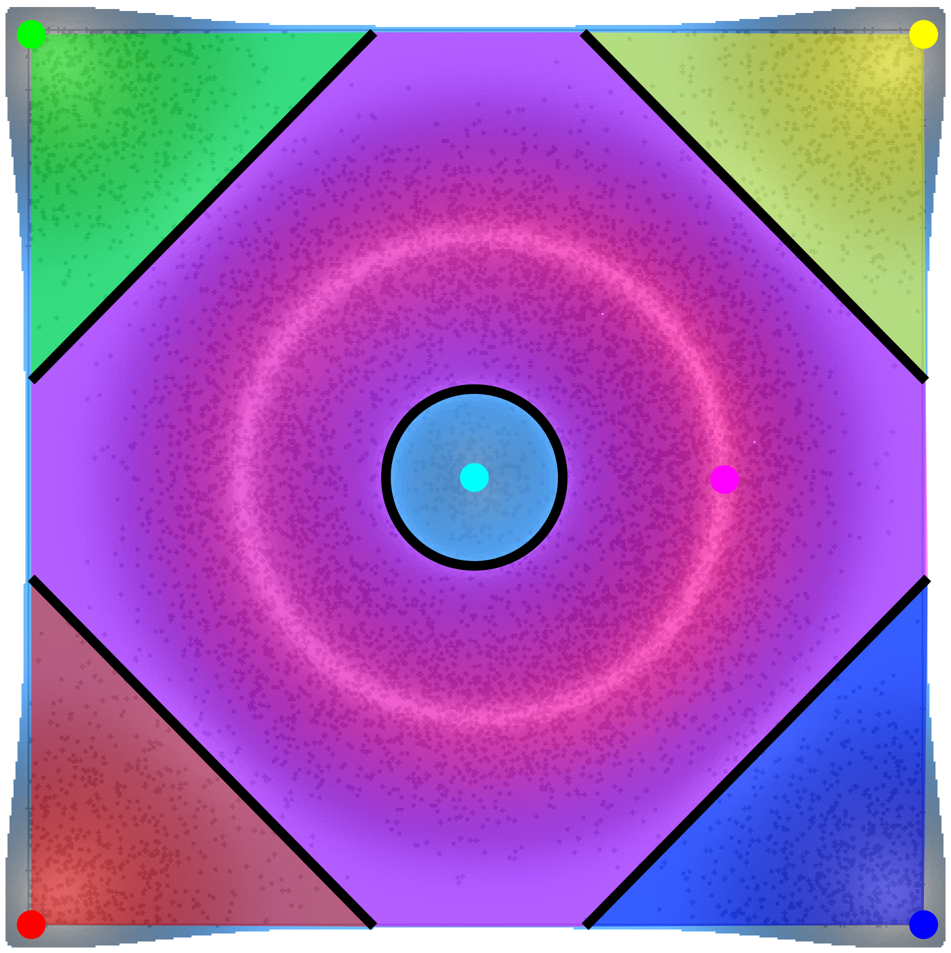
Mode-Seeking Paradigm

- Assume the data points are sampled from some unknown probability distribution
- Partition the data according to the basins of attraction of the peaks of the density



Mode-Seeking Paradigm

- Assume the data points are sampled from some unknown probability distribution
- Partition the data according to the basins of attraction of the peaks of the density

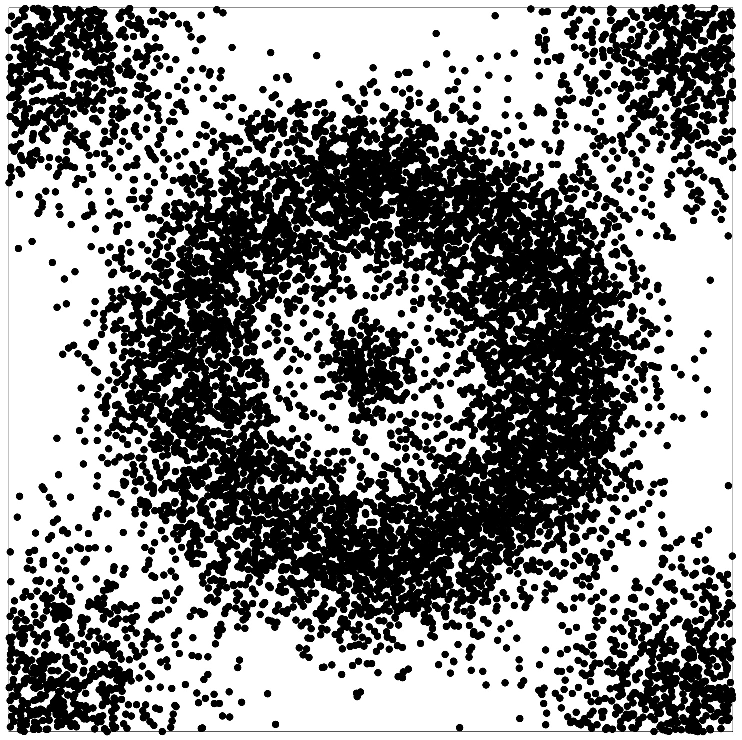


Hill-Climbing Schemes

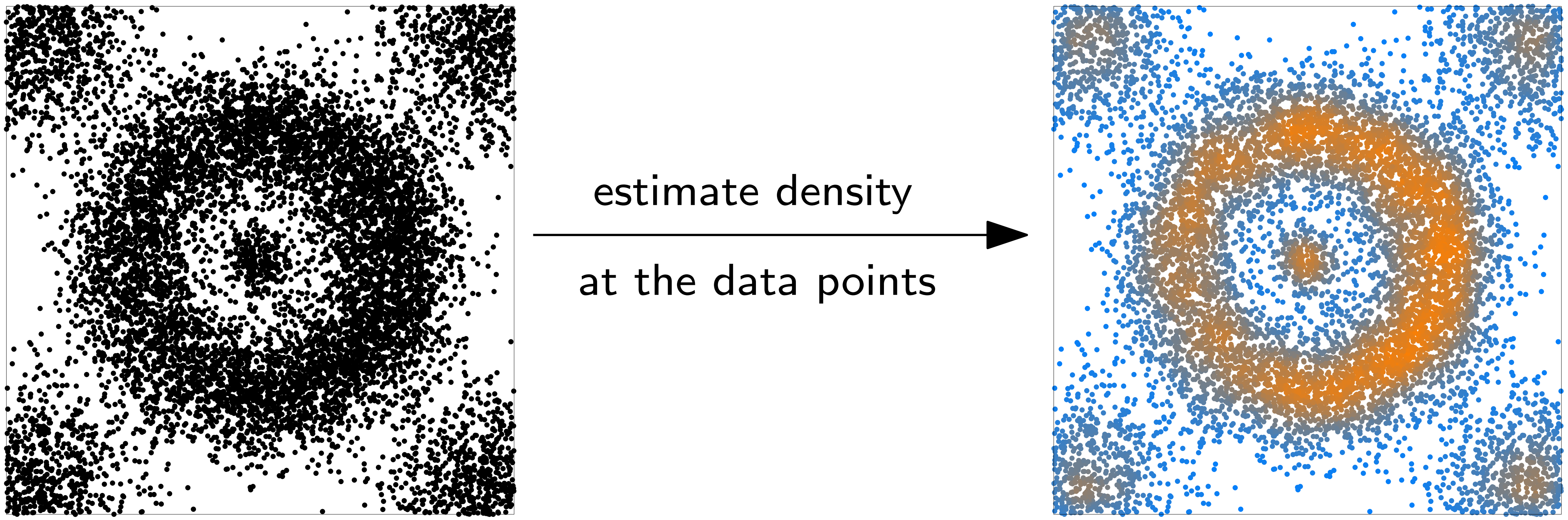
- **Iterative**, e.g. D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(5):603619, May 2002.

- **Non-iterative**, e.g. W. L. Koontz, P. M. Narendra, and K. Fukunaga. A graph-theoretic approach to nonparametric cluster analysis. *IEEE Trans. on Computers*, 24:936944, September 1976.

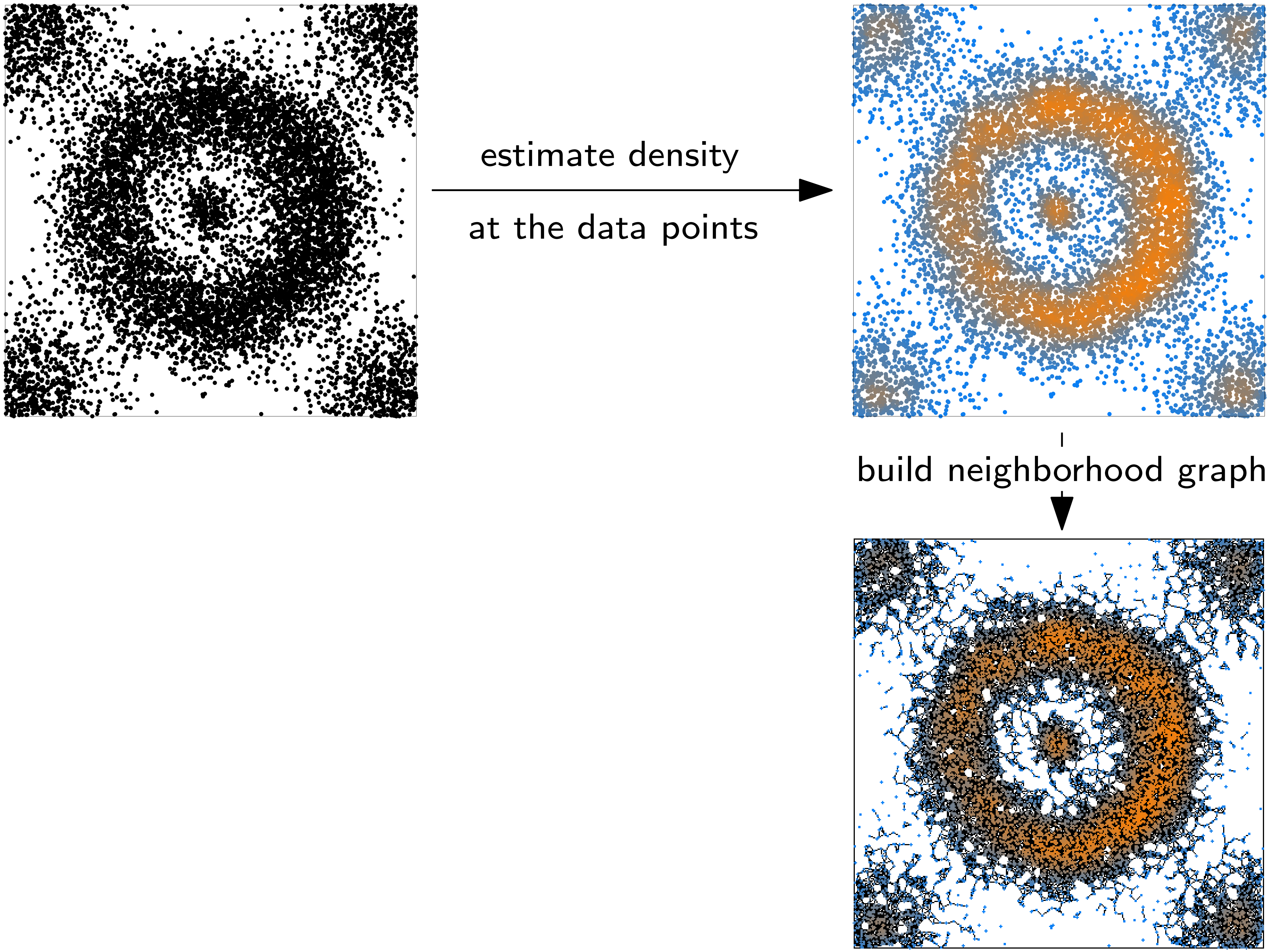
[Koontz, Narendra, Fukunaga'76] in a Nutshell



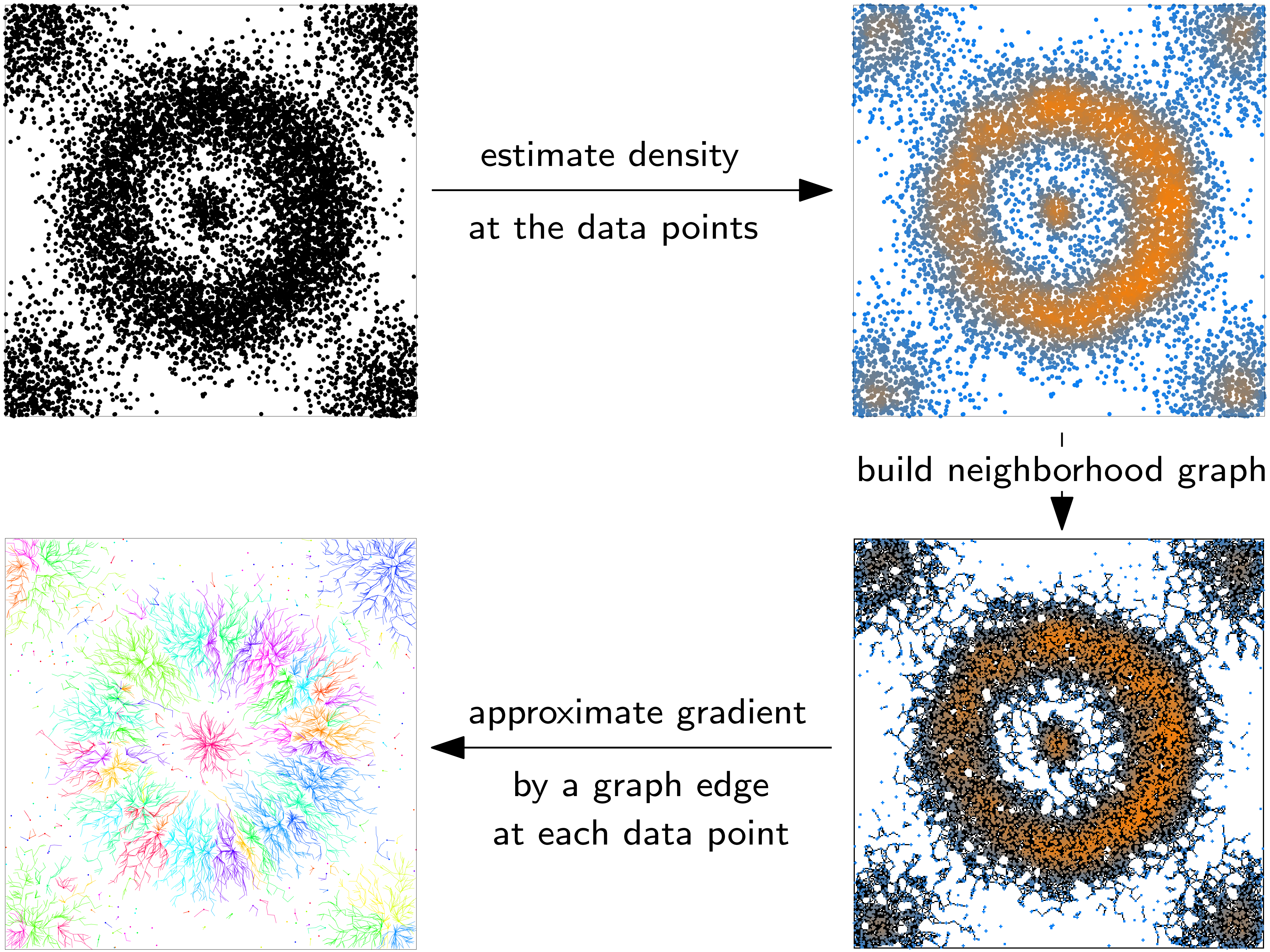
[Koontz, Narendra, Fukunaga'76] in a Nutshell



[Koontz, Narendra, Fukunaga'76] in a Nutshell

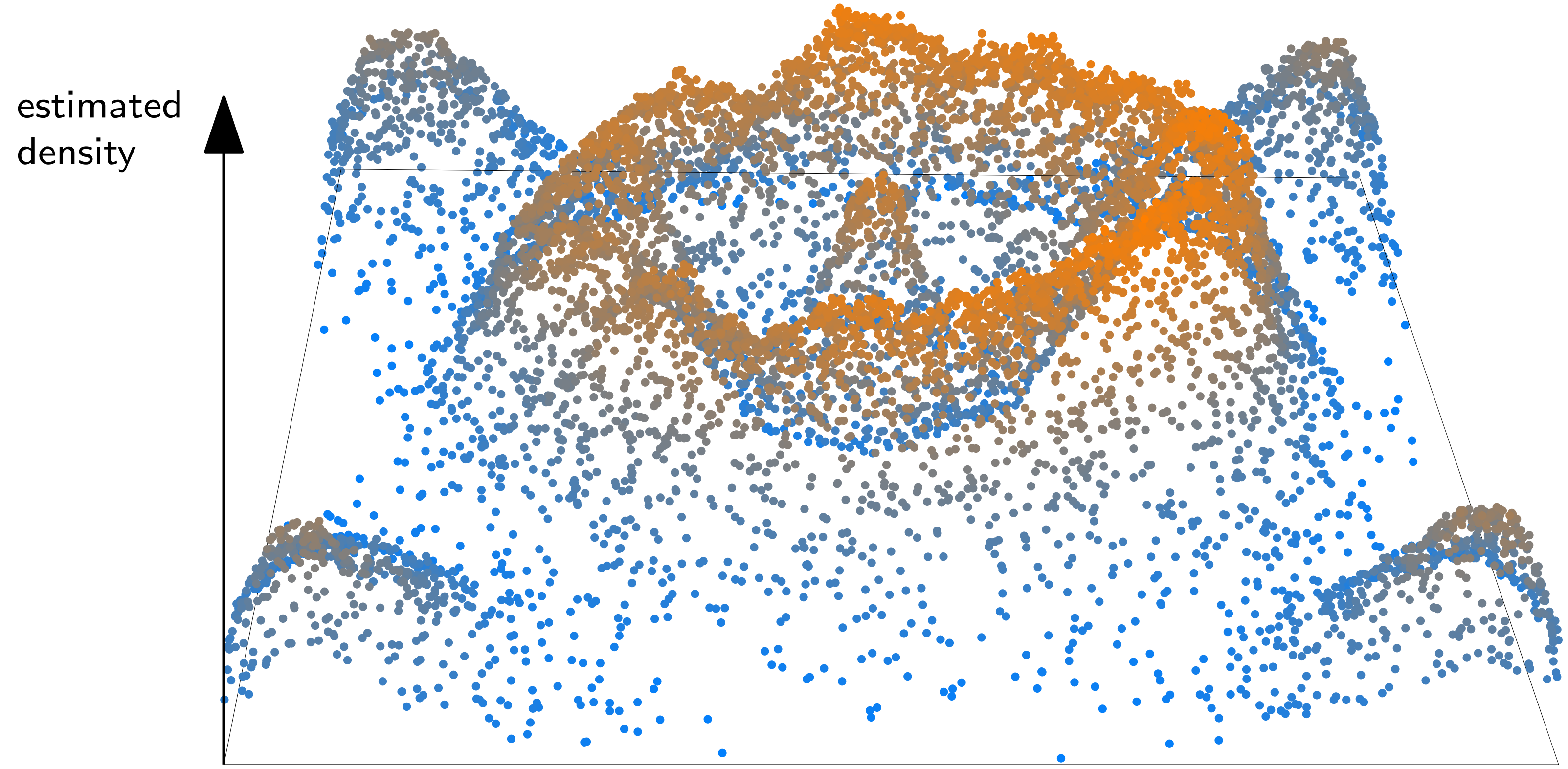


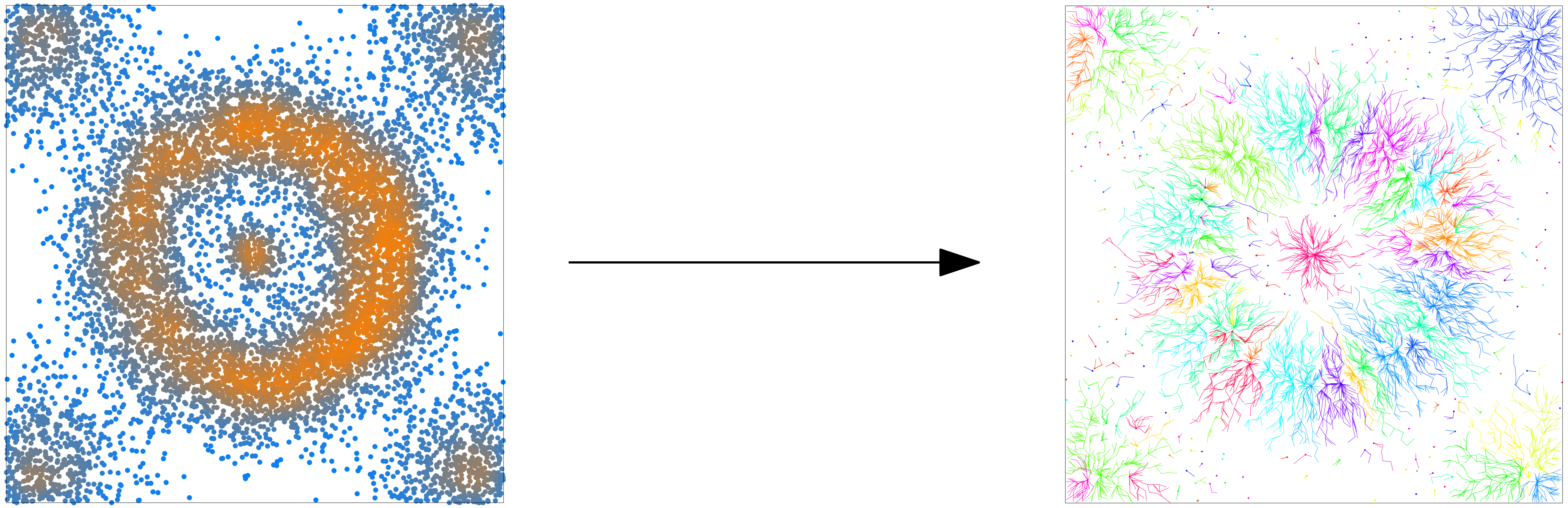
[Koontz, Narendra, Fukunaga'76] in a Nutshell



Why things are likely to go ill

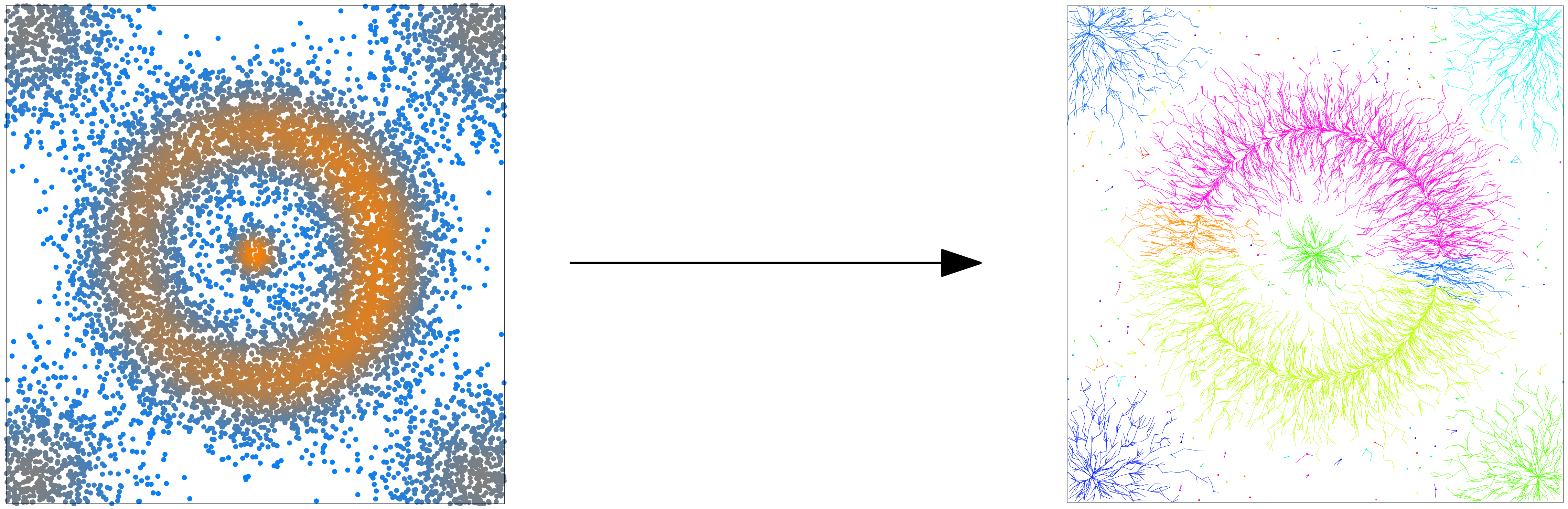
- Noisy estimator





Why things are likely to go ill

- Noisy estimator
- Neighborhood graph



Why things are likely to go ill

- Noisy estimator
- Neighborhood graph

Solutions:

1. **Be proactive**: smooth-out estimator before clustering, a la Mean-Shift

   $\rightarrow$ how much smoothing is needed?

   $\rightarrow$ does not solve the neighborhood graph issue

Why things are likely to go ill

- Noisy estimator
- Neighborhood graph

Solutions:

1. **Be proactive**: smooth-out estimator before clustering, a la Mean-Shift

   $\rightarrow$ how much smoothing is needed?

   $\rightarrow$ does not solve the neighborhood graph issue

2. **Be reactive**: merge clusters after clustering, to regain some stability

   $\rightarrow$ repeat mode-seeking until convergence (Medoid-Shift [SKK'07])

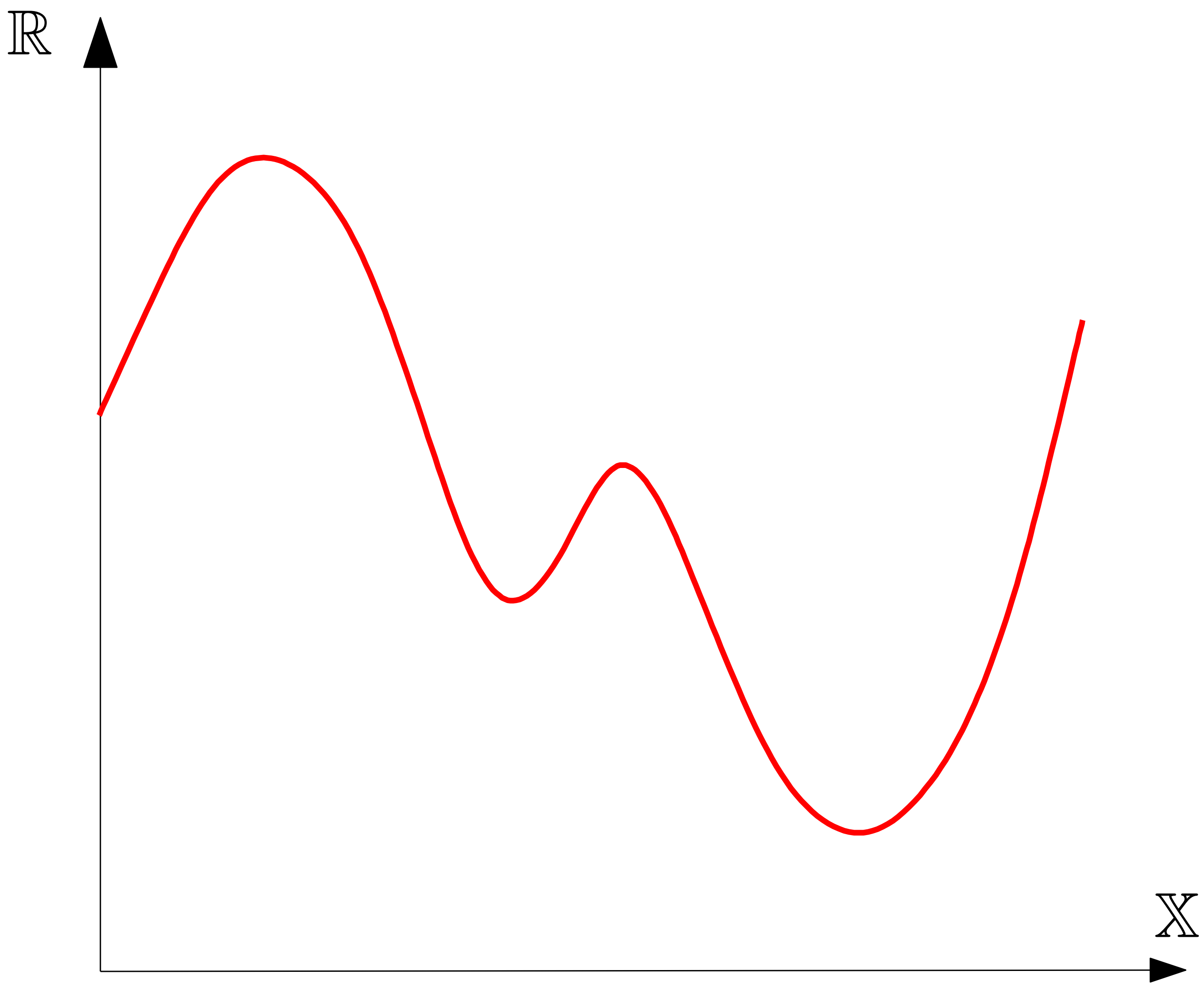
   $\rightarrow$ use topological persistence to guide a single-pass merging step

Enter Topological Persistence...

Persistence for Mode Seeking

Given a probability density f:

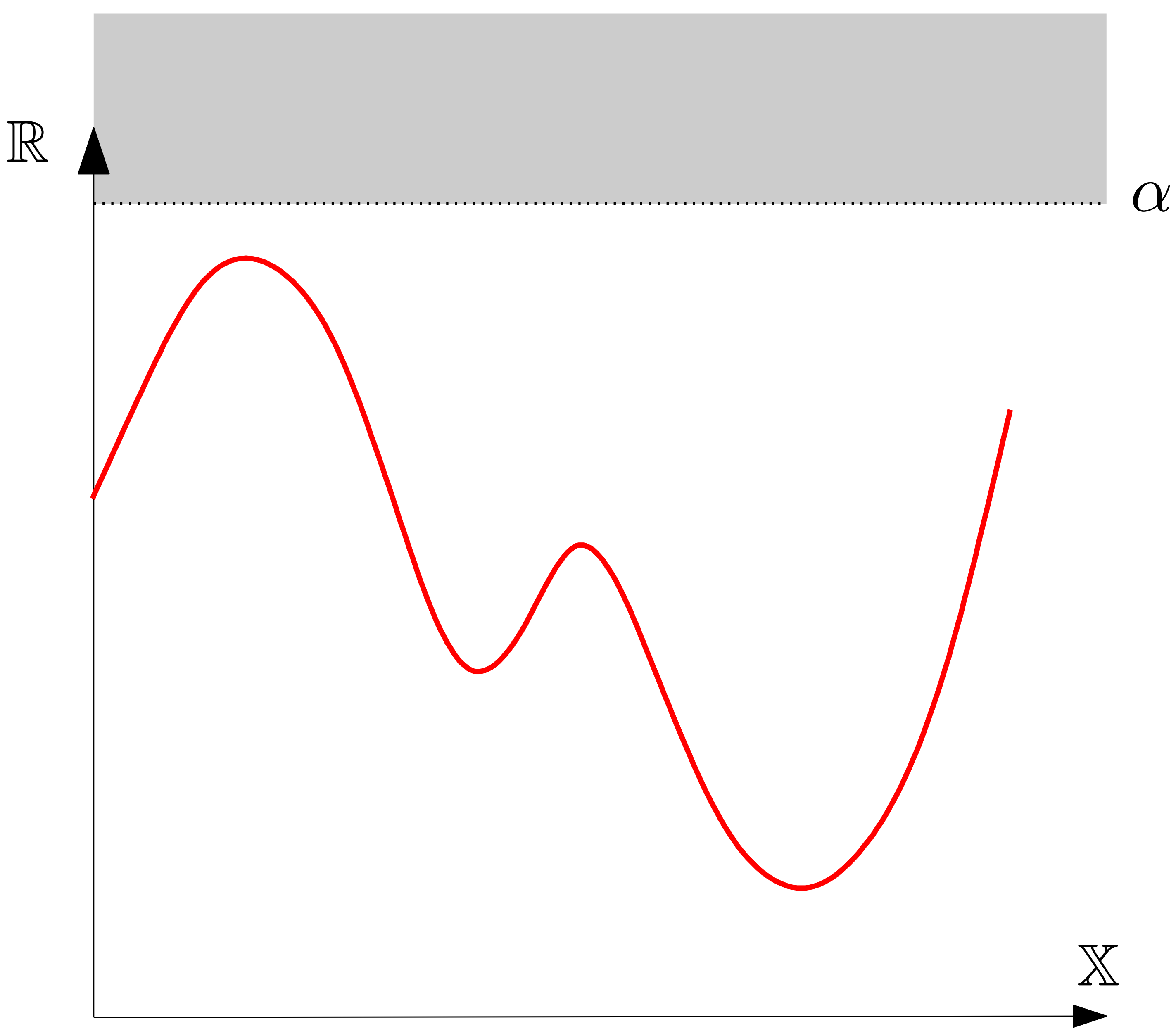
- Nested family (filtration) of superlevel-sets $f^{-1}([\alpha, +\infty))$ for $\alpha = +\infty$ to $-\infty$.
- Track evolution of topology throughout the family.



Persistence for Mode Seeking

Given a probability density f:

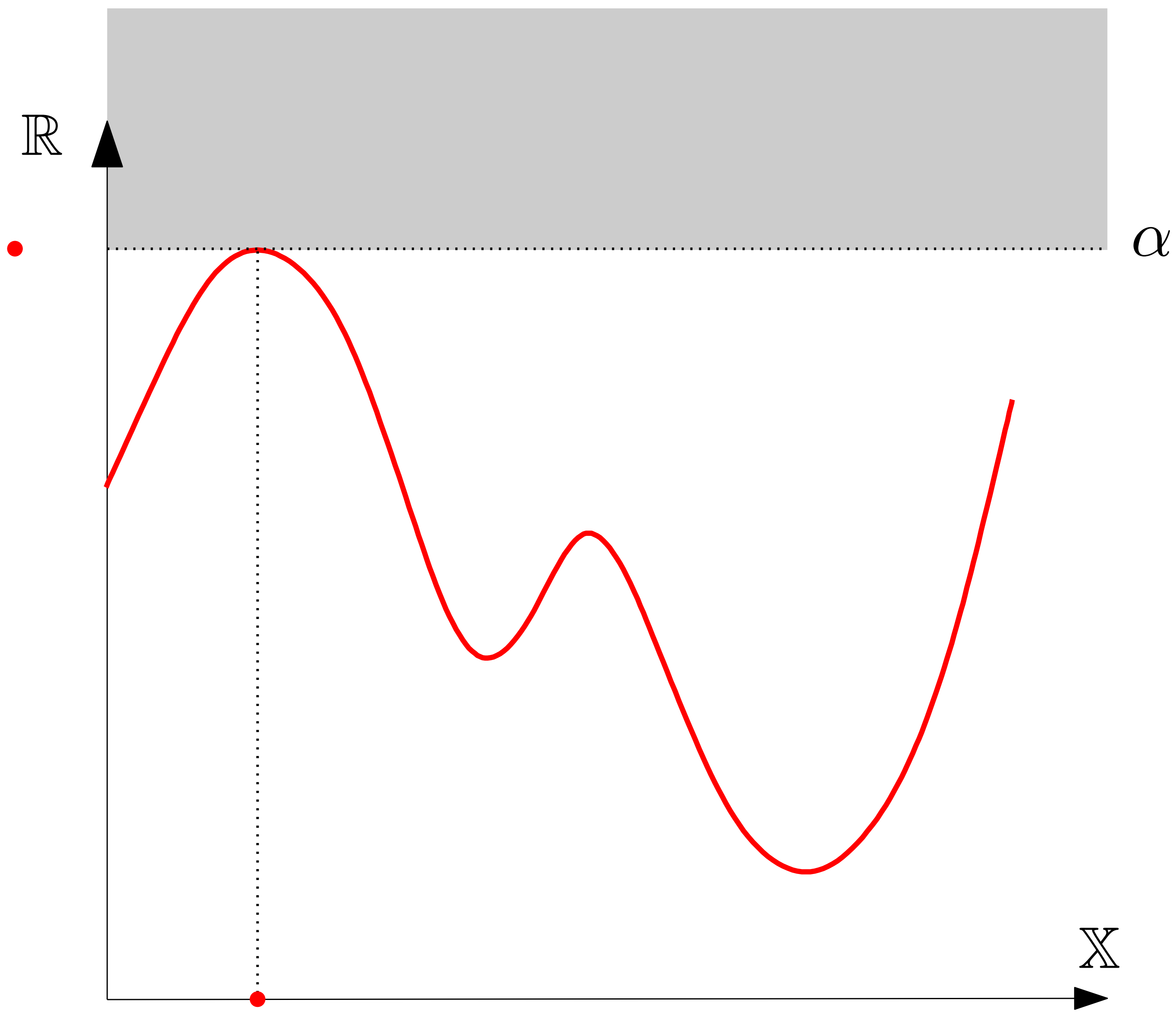
- Nested family (filtration) of superlevel-sets $f^{-1}([\alpha, +\infty))$ for $\alpha = +\infty$ to $-\infty$.
- Track evolution of topology throughout the family.



Persistence for Mode Seeking

Given a probability density f:

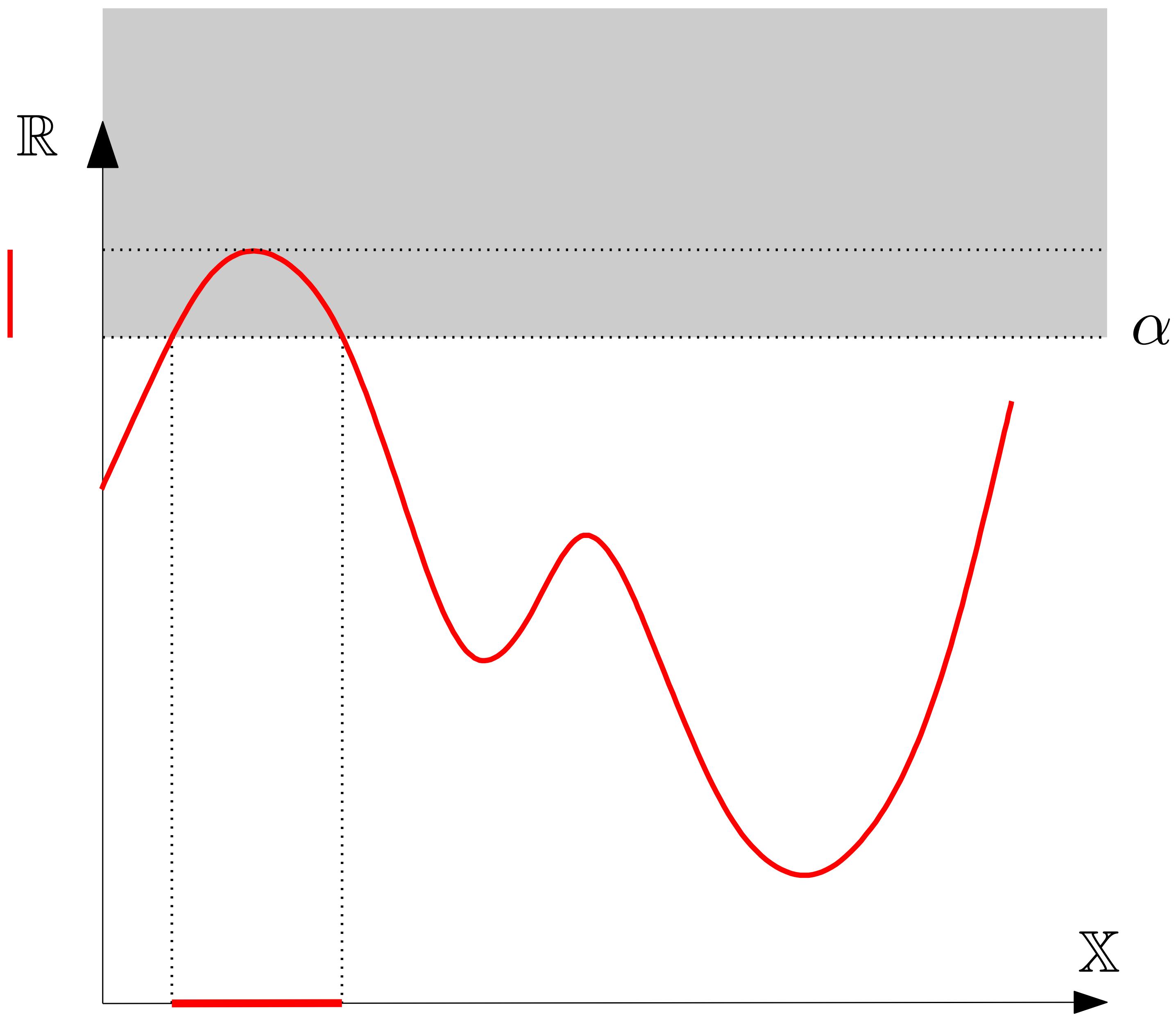
- Nested family (filtration) of superlevel-sets $f^{-1}([\alpha, +\infty))$ for $\alpha = +\infty$ to $-\infty$.
- Track evolution of topology throughout the family.



Persistence for Mode Seeking

Given a probability density f:

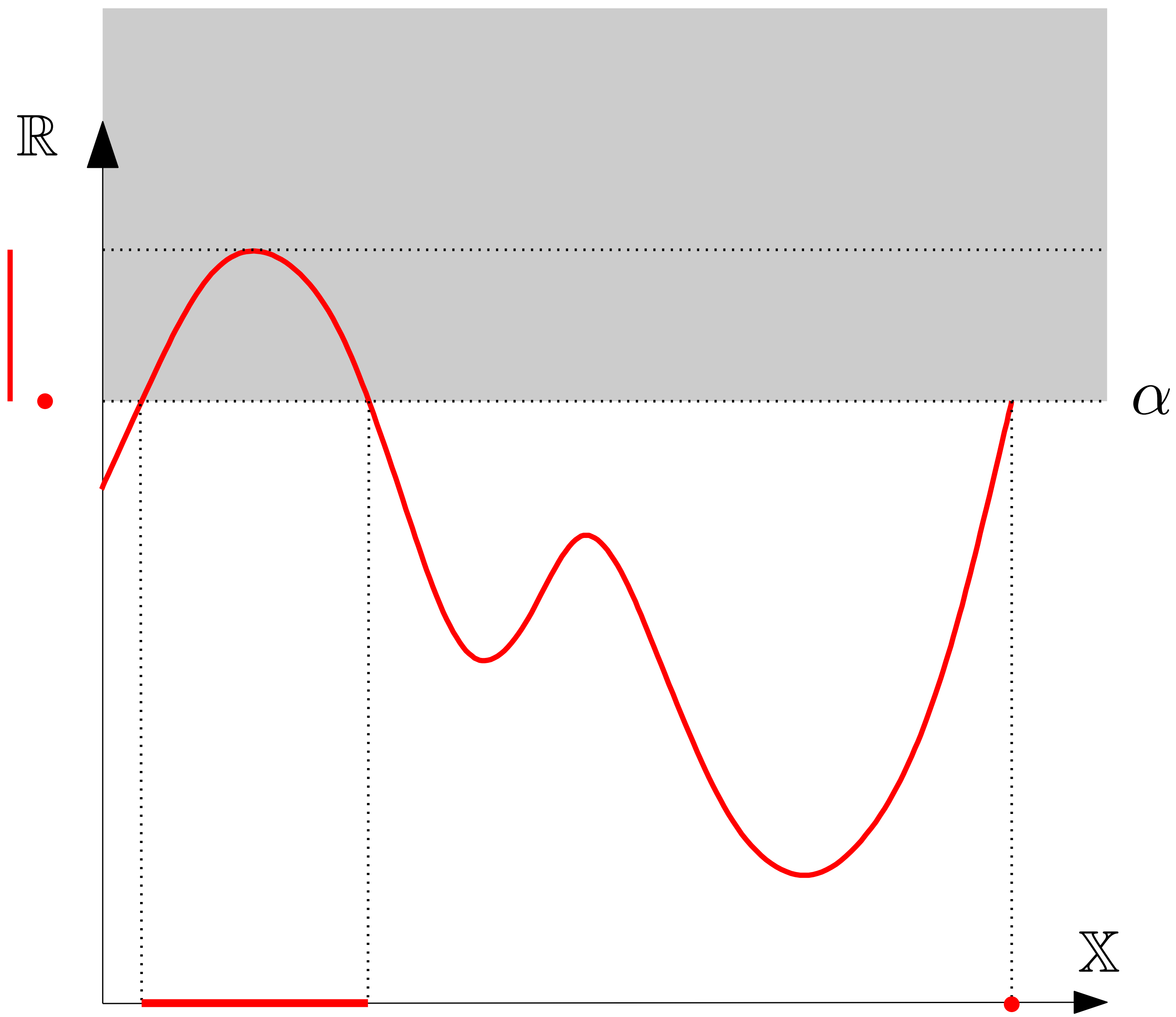
- Nested family (filtration) of superlevel-sets $f^{-1}([\alpha, +\infty))$ for $\alpha = +\infty$ to $-\infty$.
- Track evolution of topology throughout the family.



Persistence for Mode Seeking

Given a probability density f:

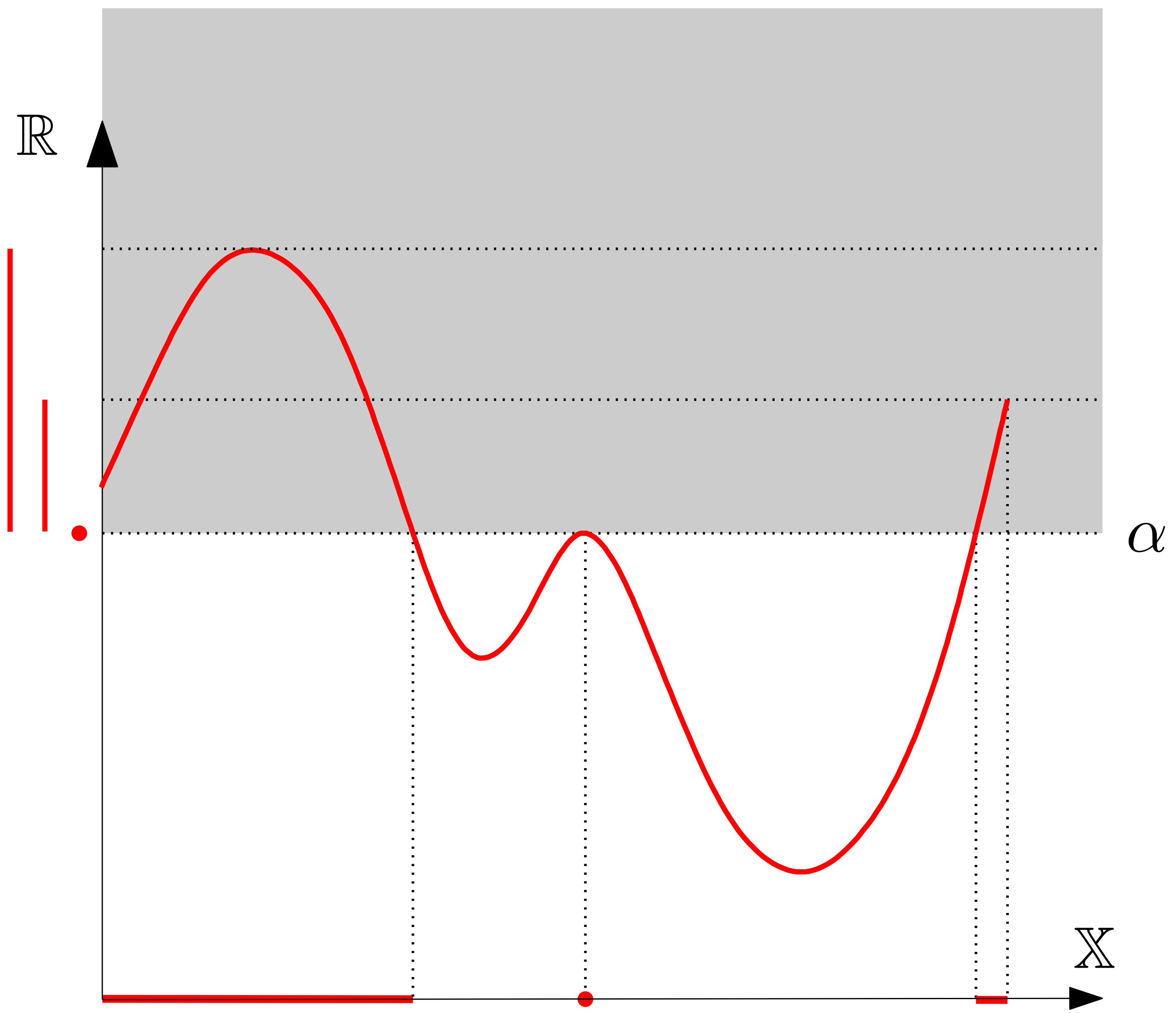
- Nested family (filtration) of superlevel-sets $f^{-1}([\alpha, +\infty))$ for $\alpha = +\infty$ to $-\infty$.
- Track evolution of topology throughout the family.



Persistence for Mode Seeking

Given a probability density f:

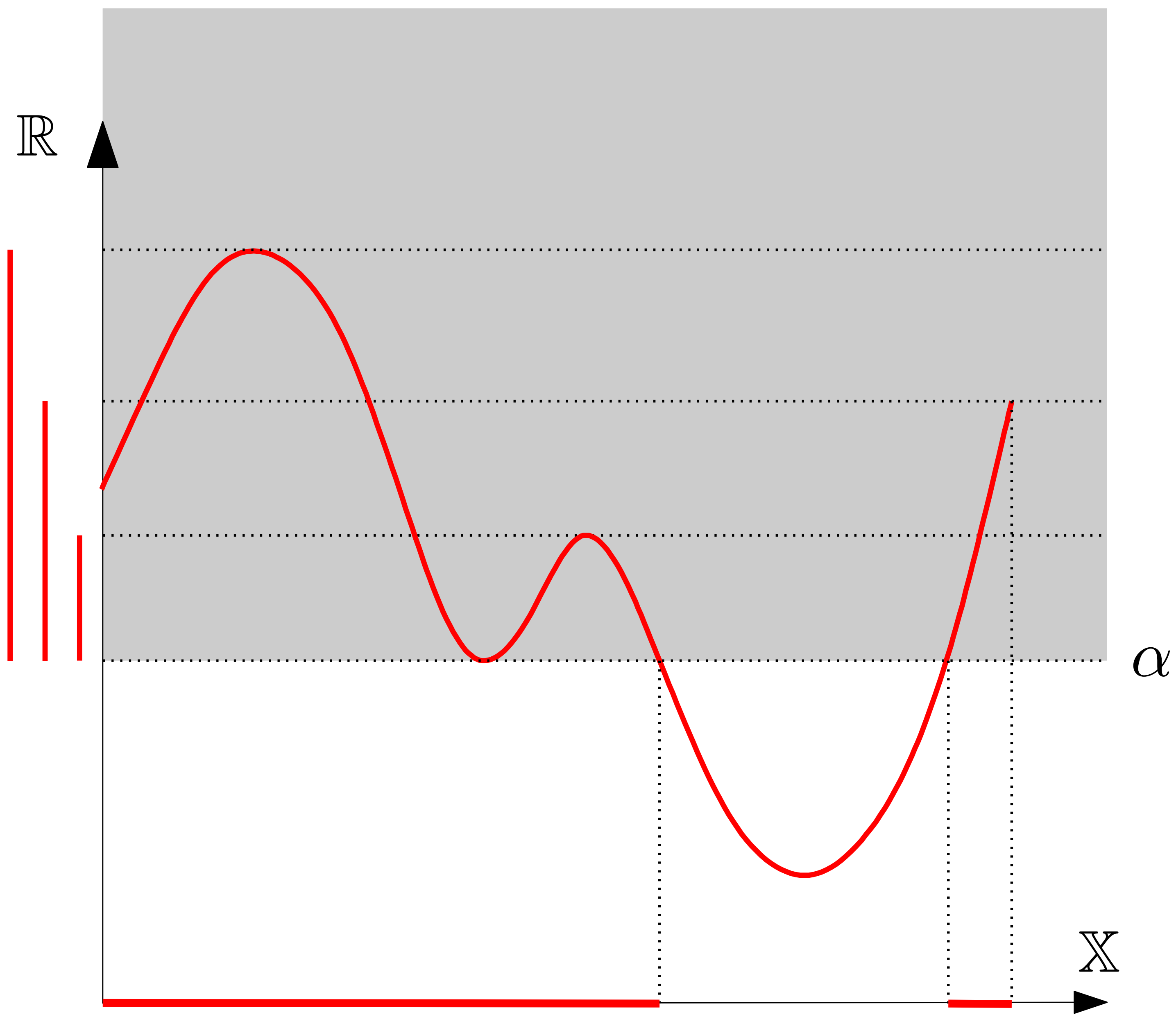
- Nested family (filtration) of superlevel-sets $f^{-1}([\alpha, +\infty))$ for $\alpha = +\infty$ to $-\infty$.
- Track evolution of topology throughout the family.



Persistence for Mode Seeking

Given a probability density f:

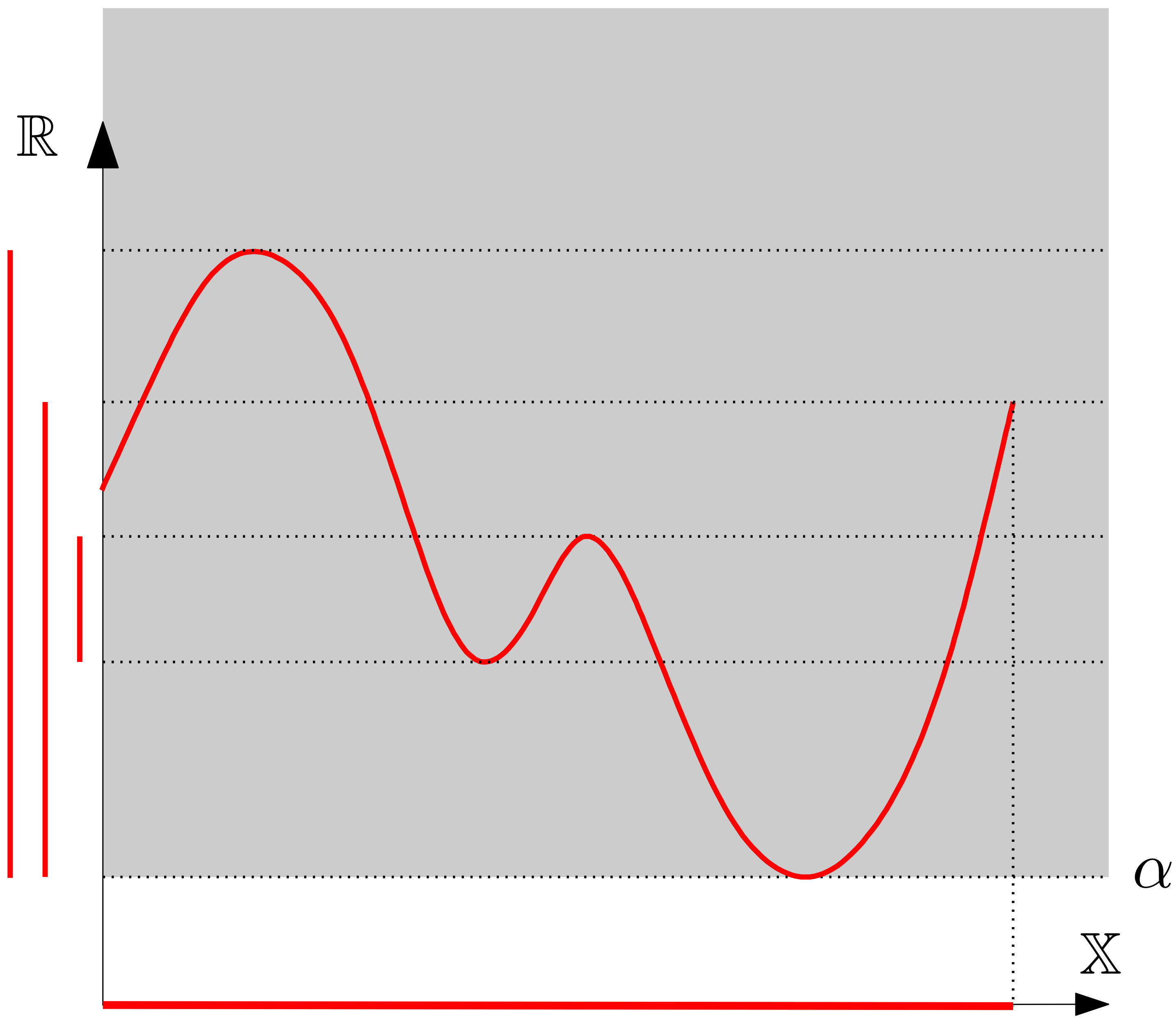
- Nested family (filtration) of superlevel-sets $f^{-1}([\alpha, +\infty))$ for $\alpha = +\infty$ to $-\infty$.
- Track evolution of topology throughout the family.



Persistence for Mode Seeking

Given a probability density f:

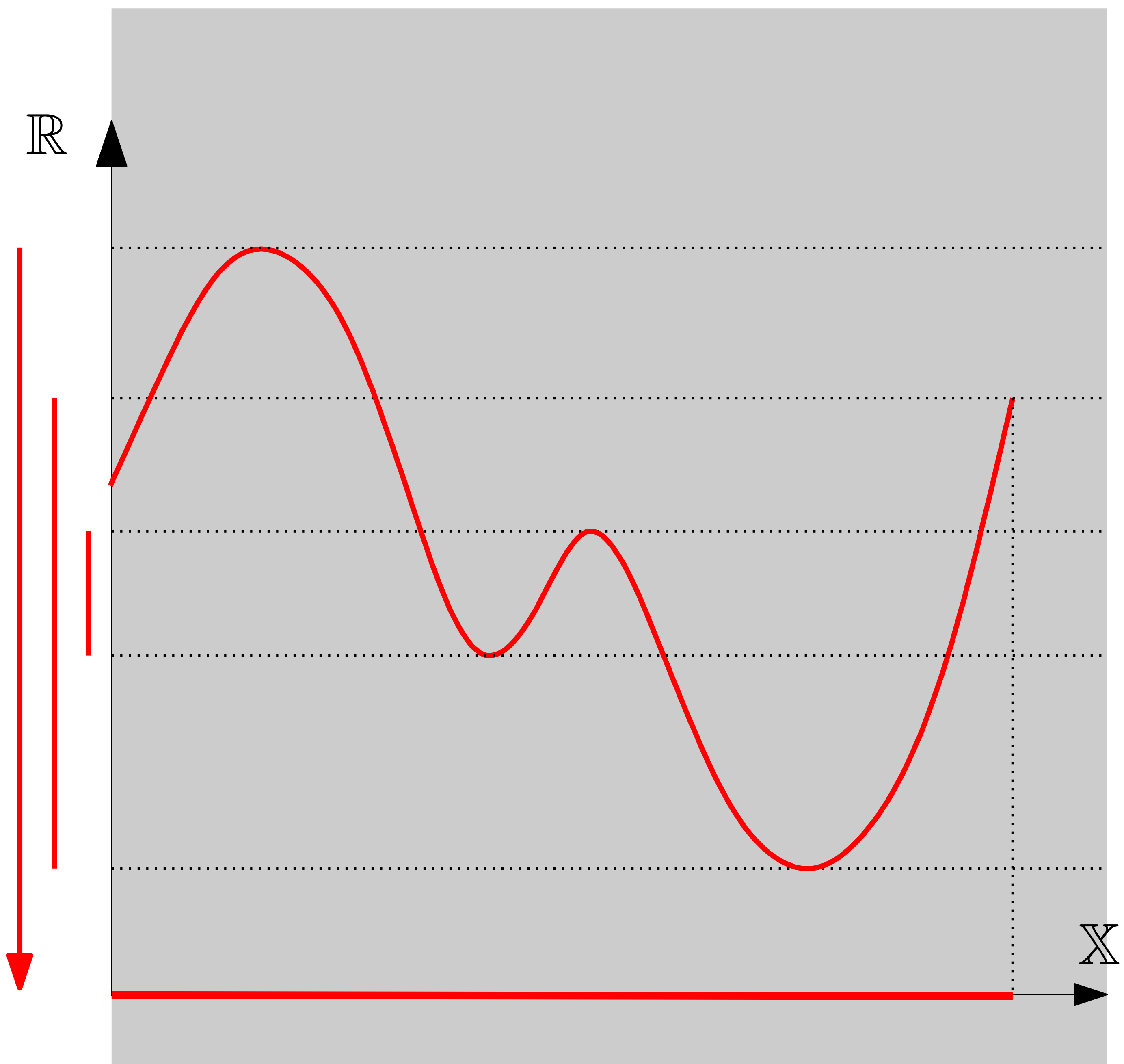
- Nested family (filtration) of superlevel-sets $f^{-1}([\alpha, +\infty))$ for $\alpha = +\infty$ to $-\infty$.
- Track evolution of topology throughout the family.



Persistence for Mode Seeking

Given a probability density f:

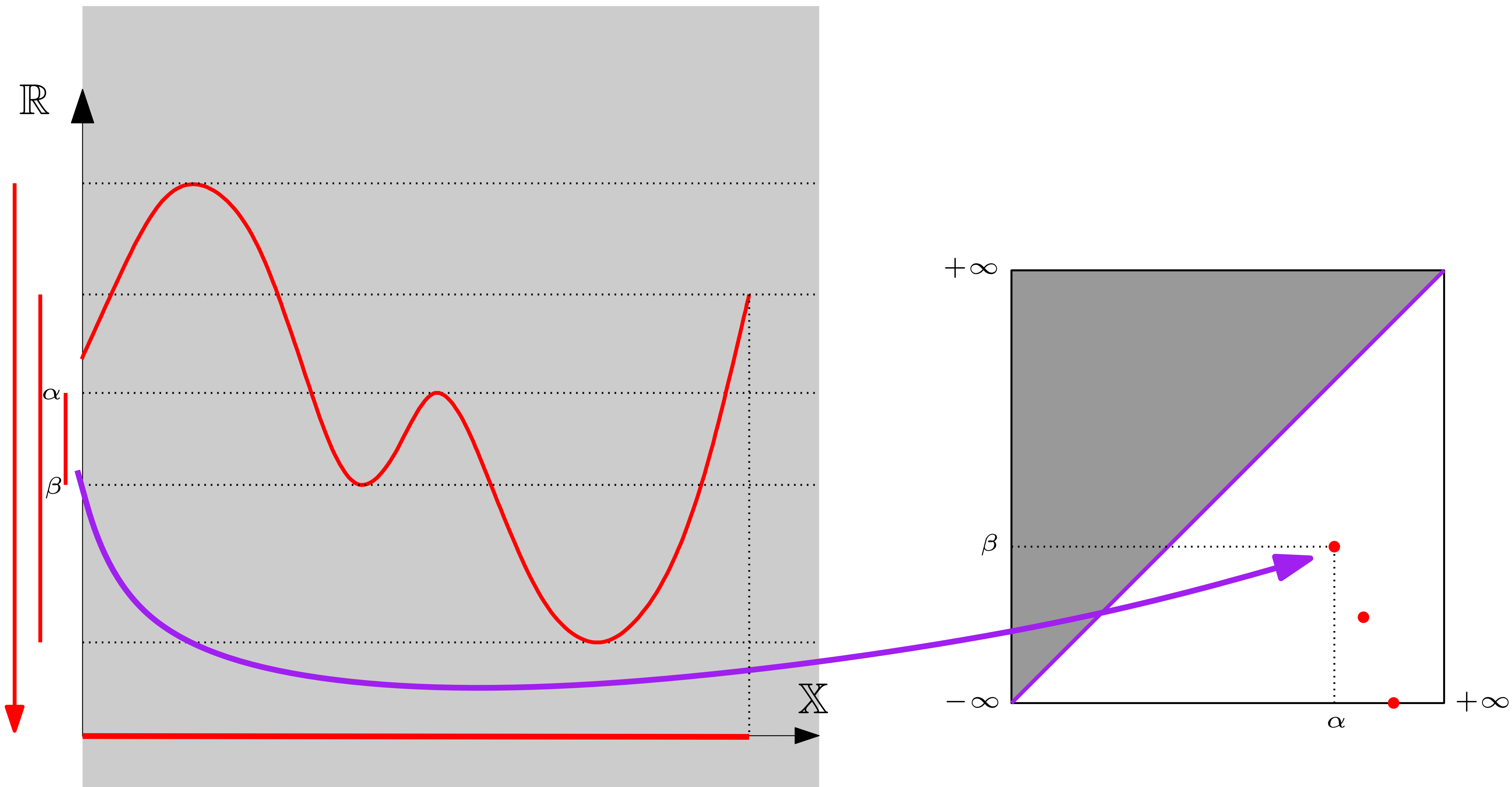
- Nested family (filtration) of superlevel-sets $f^{-1}([\alpha, +\infty))$ for $\alpha = +\infty$ to $-\infty$.
- Track evolution of topology throughout the family.
- Finite set of intervals (barcode) encodes births/deaths of topological features.



Persistence for Mode Seeking

Given a probability density f:

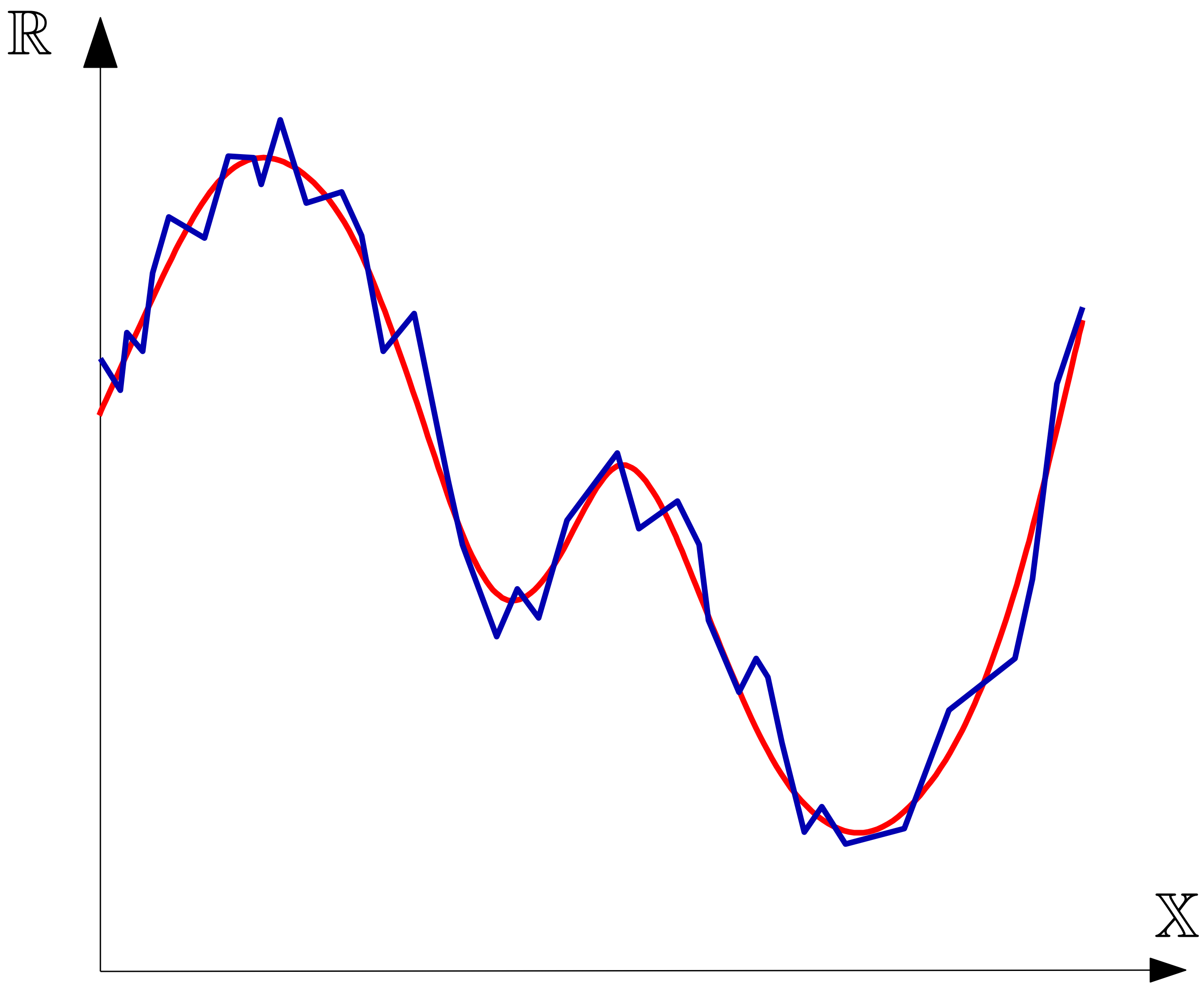
- Nested family (filtration) of superlevel-sets $f^{-1}([\alpha, +\infty))$ for $\alpha = +\infty$ to $-\infty$.
- Track evolution of topology throughout the family.
- Finite set of intervals (barcode) encodes births/deaths of topological features.

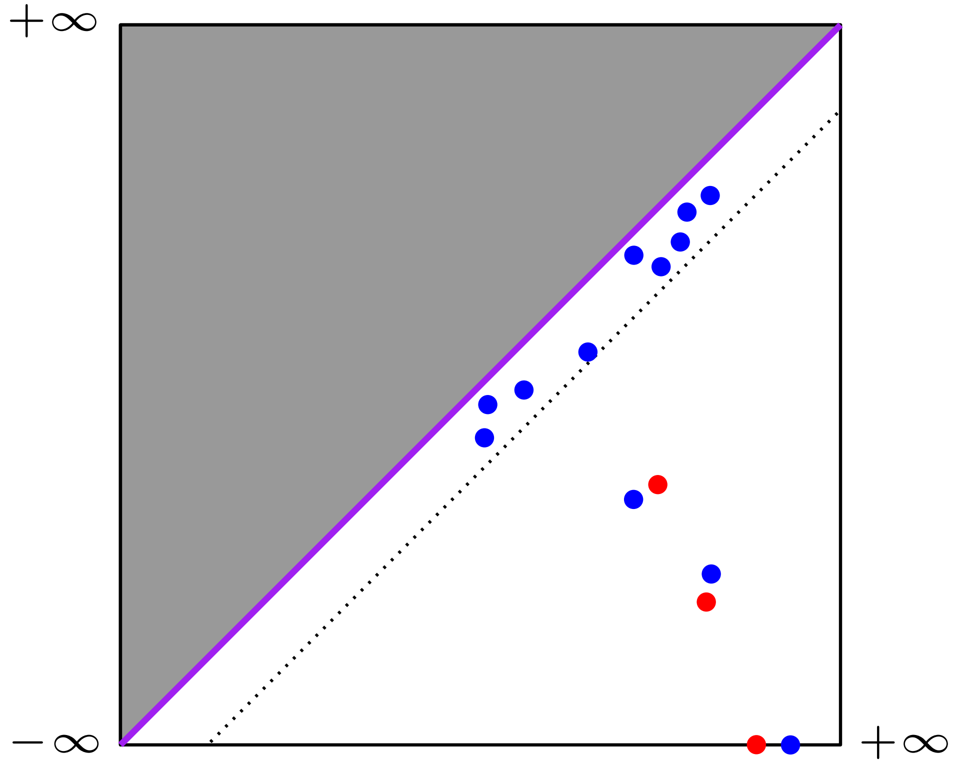


Persistence for Mode Seeking

Given an estimator $\hat{f}$:

Stability Theorem $\Rightarrow \mathrm{d}_B^{\infty}(\mathrm{Dg}\, f, \mathrm{Dg}\, \hat{f}) \leq \|f - \hat{f}\|_{\infty}$.

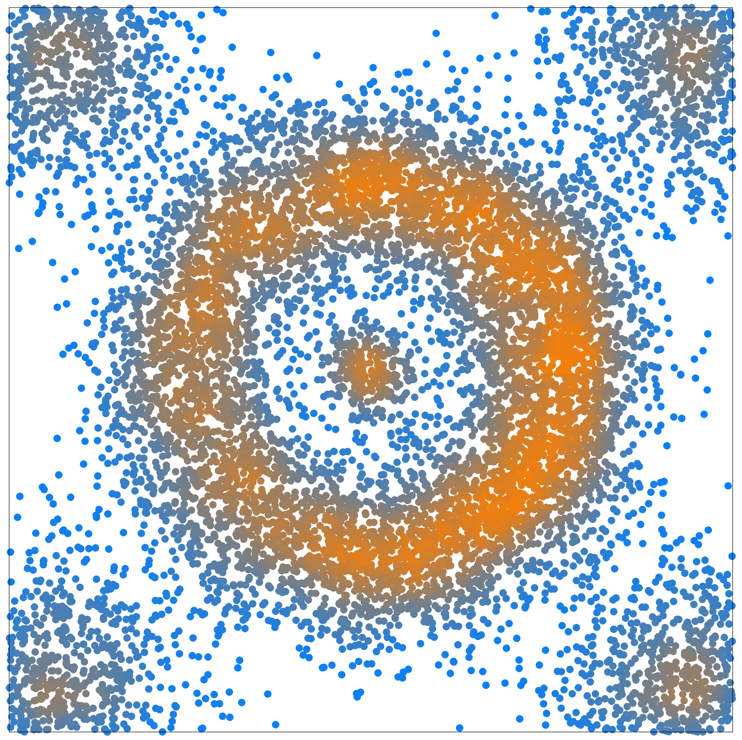




More precisely...

- Density estimator $\hat{f}$ defines an order on the point cloud

  (sort data points by **decreasing** estimated density values)



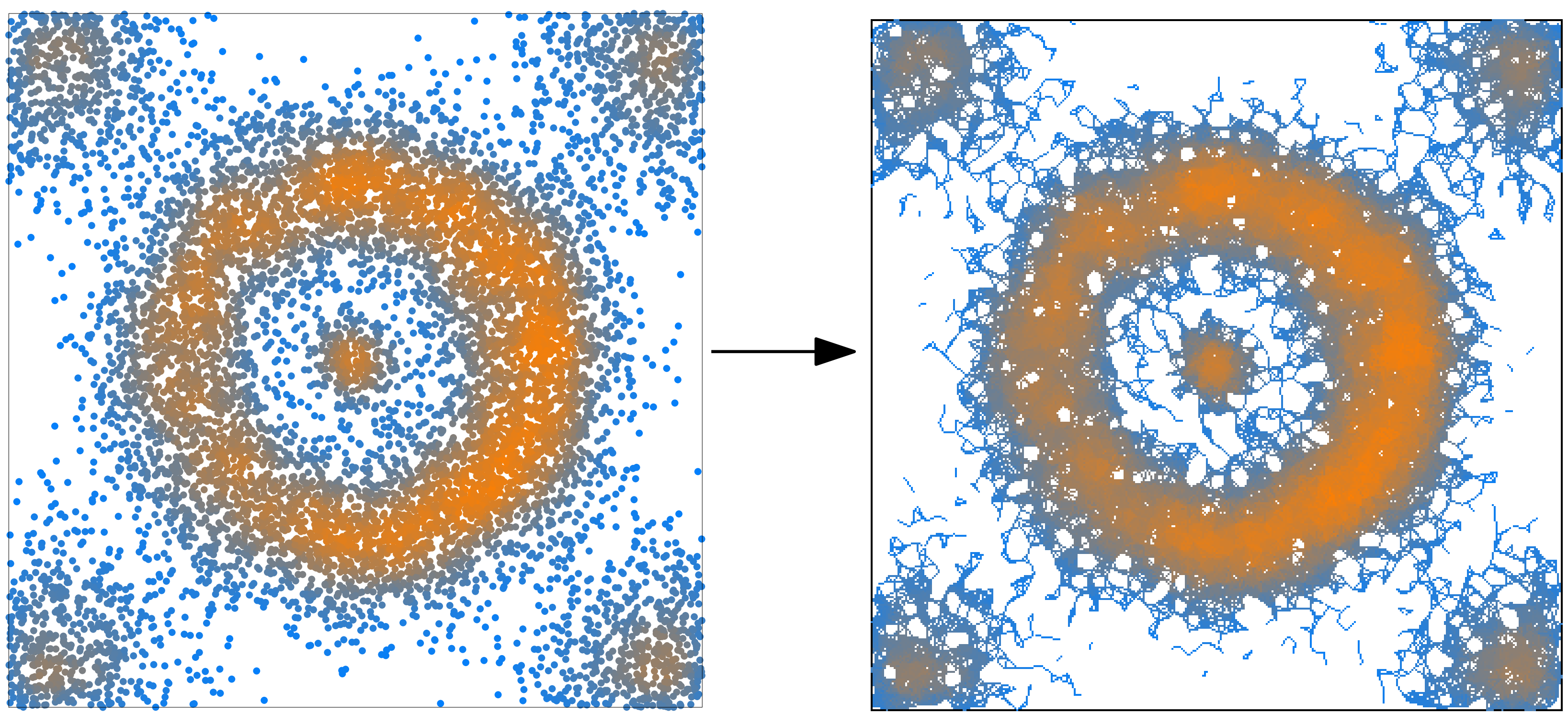
More precisely...

- Density estimator $\hat{f}$ defines an order on the point cloud

  (sort data points by **decreasing** estimated density values)

- Extend order to the graph edges $\rightarrow$ *upper-star filtration*

  $(\hat{f}([u,v]) = \min\{\hat{f}(u),\ \hat{f}(v)\})$



More precisely...

- Density estimator $\hat{f}$ defines an order on the point cloud

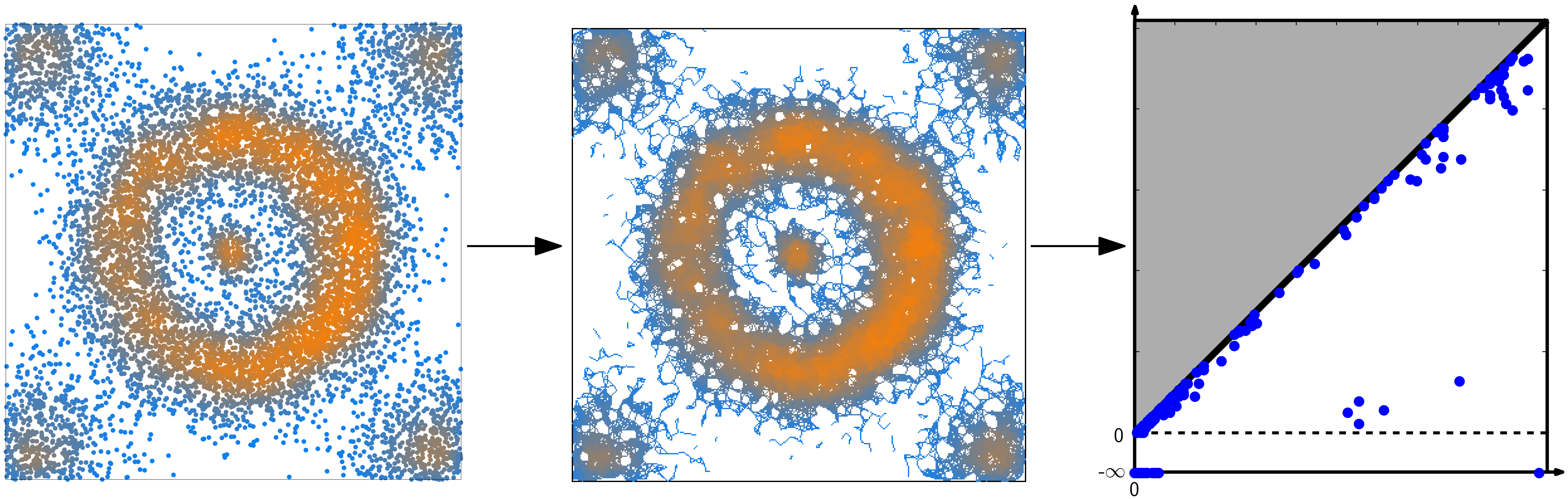
  (sort data points by **decreasing** estimated density values)

- Extend order to the graph edges $\rightarrow$ *upper-star filtration*

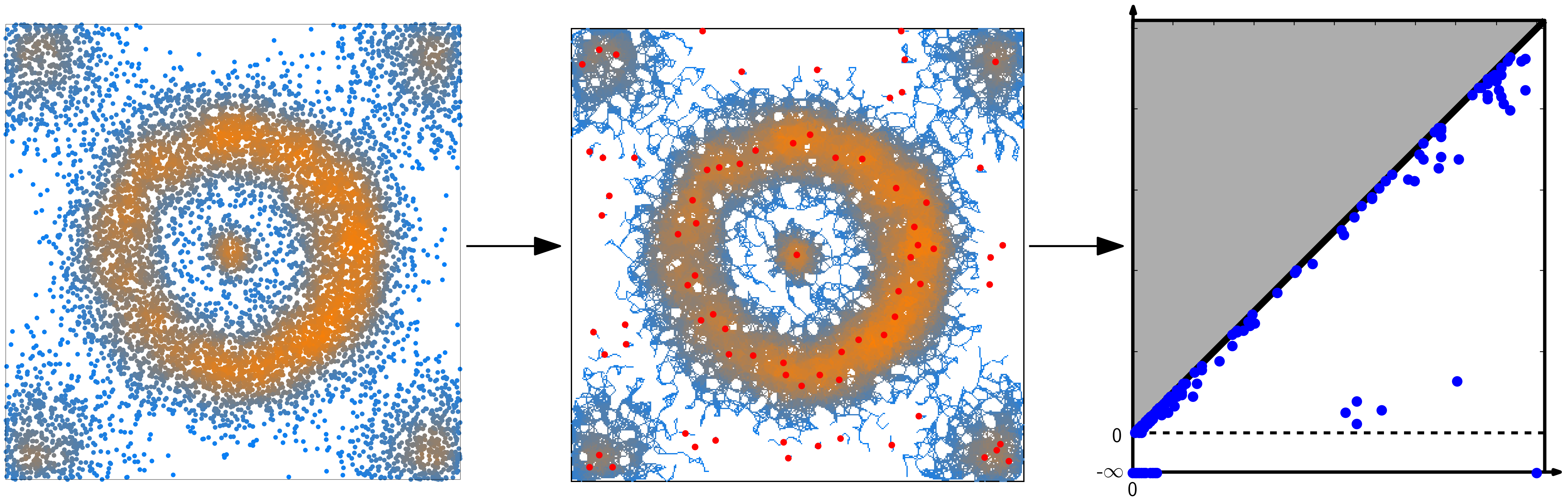
  $(\hat{f}([u,v]) = \min\{\hat{f}(u),\ \hat{f}(v)\})$

- Compute the 0-dimensional persistence diagram of this filtration

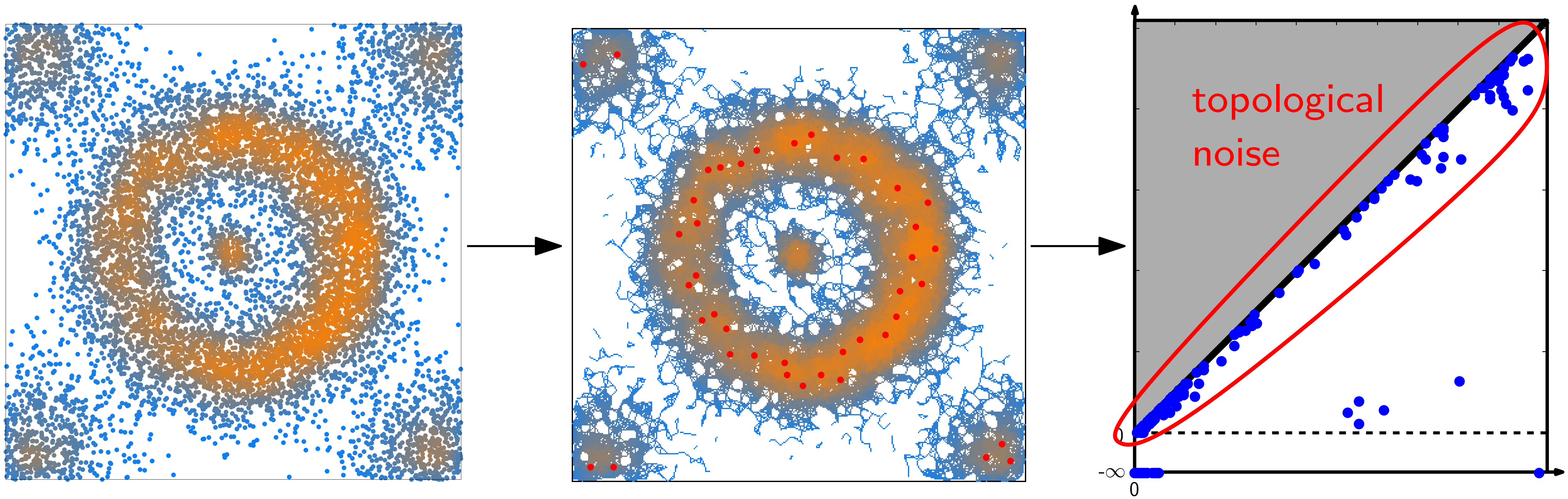
  (apply 0-dimensional persistence algorithm $\rightarrow$ union-find data structure)



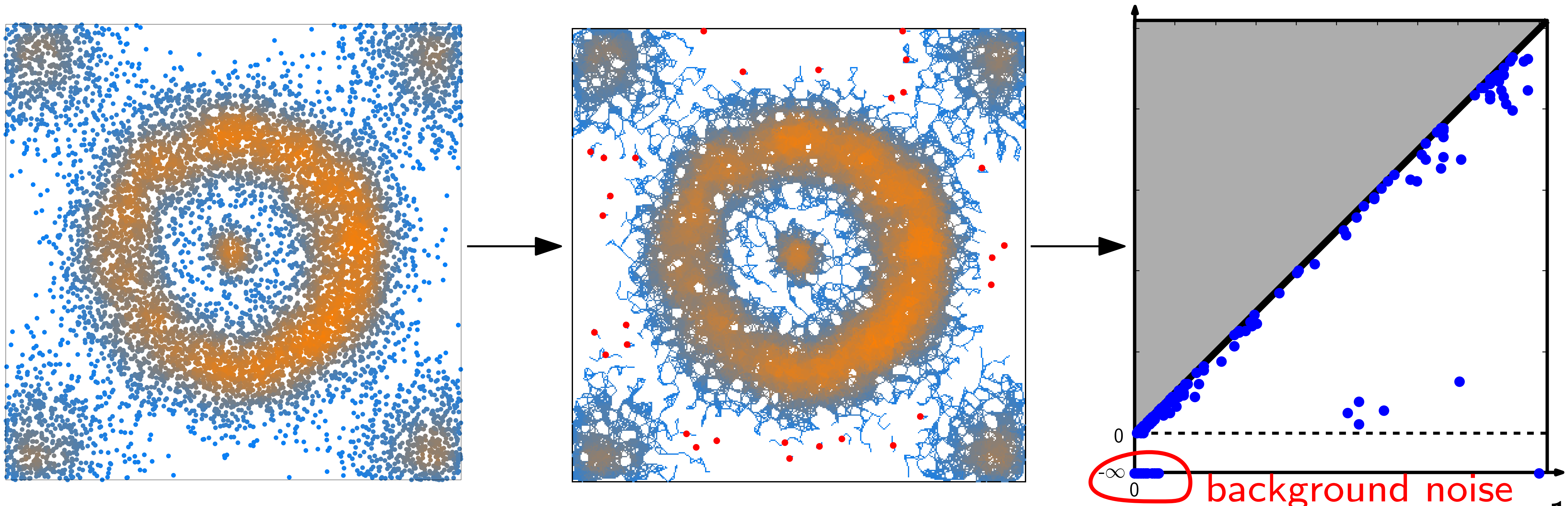
Estimating the Correct Number of Clusters



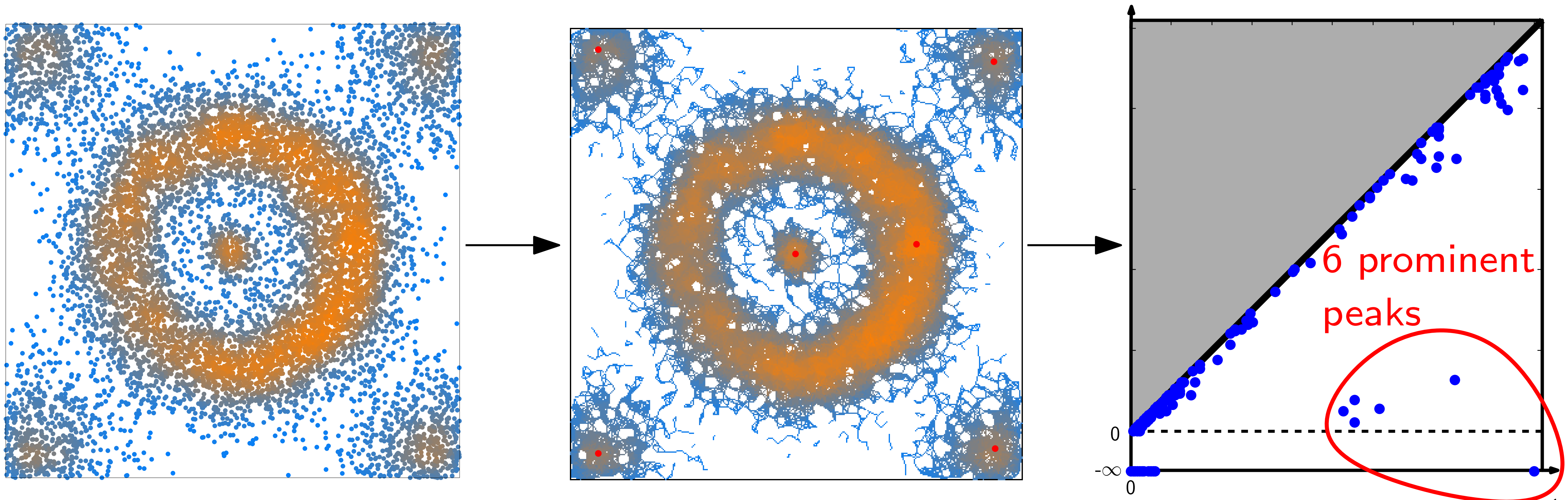
Estimating the Correct Number of Clusters



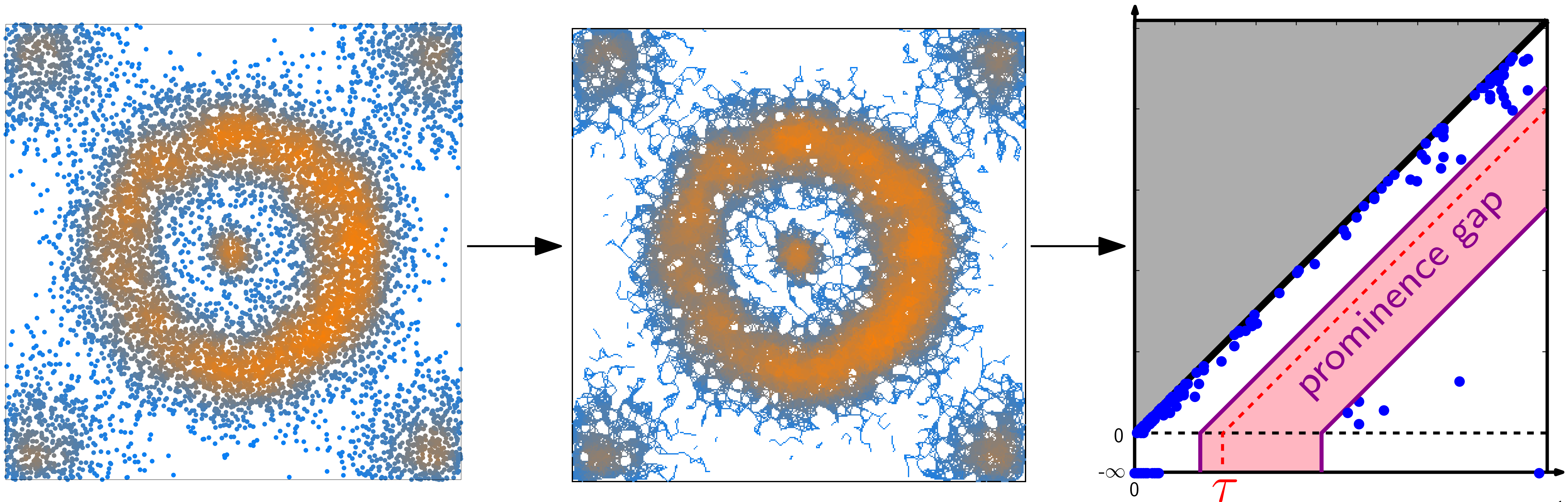
Estimating the Correct Number of Clusters



Estimating the Correct Number of Clusters



Estimating the Correct Number of Clusters



Estimating the Correct Number of Clusters

Hypotheses:

- $\mathbb{X}$ a Riemmaniann manifold with positive convexity radius $\varrho(\mathbb{X})$,
- $f : \mathbb{X} \to \mathbb{R}$ a c-Lipschitz probability density function,

Estimating the Correct Number of Clusters

Hypotheses:

- $\mathbb{X}$ a Riemmaniann manifold with positive convexity radius $\varrho(\mathbb{X})$,
- $f : \mathbb{X} \to \mathbb{R}$ a c-Lipschitz probability density function,
- P a finite set of n points of $\mathbb{X}$ sampled i.i.d. according to f,
- $\hat{f} : P \to \mathbb{R}$ a density estimator such that $\eta := \max_{p \in P} |\hat{f}(p) - f(p)| < \Pi/5$,
- $G = (P, E)$ the δ-Rips graph for some positive $\delta < \min\left\{\varrho(\mathbb{X}),\ \frac{\Pi - 5\eta}{5c}\right\}$.

Note: Π is the prominence of the least prominent peak of f

Estimating the Correct Number of Clusters

Hypotheses:

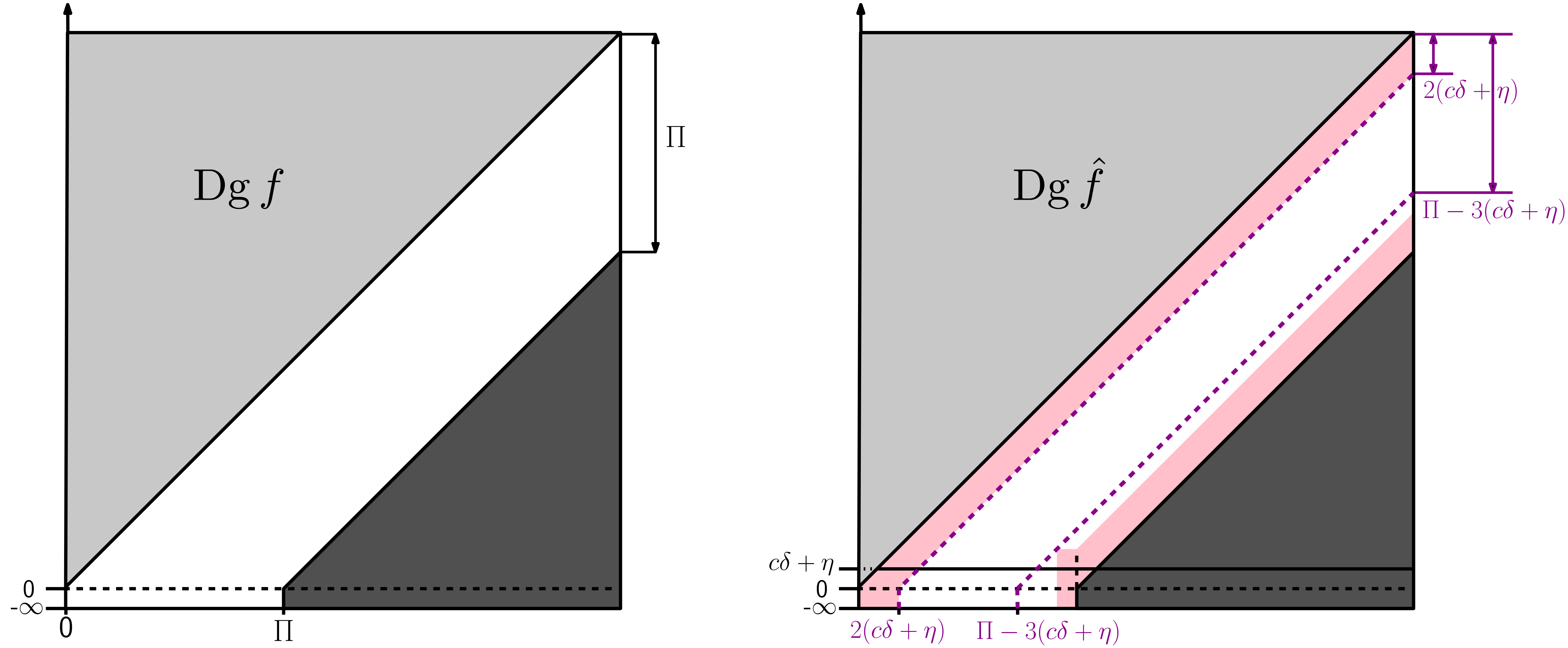
- $\mathbb{X}$ a Riemmaniann manifold with positive convexity radius $\varrho(\mathbb{X})$,
- $f : \mathbb{X} \to \mathbb{R}$ a c-Lipschitz probability density function,
- P a finite set of n points of $\mathbb{X}$ sampled i.i.d. according to f,
- $\hat{f} : P \to \mathbb{R}$ a density estimator such that $\eta := \max_{p \in P} |\hat{f}(p) - f(p)| < \Pi/5$,
- $G = (P, E)$ the δ-Rips graph for some positive $\delta < \min\left\{\varrho(\mathbb{X}),\ \frac{\Pi - 5\eta}{5c}\right\}$.

Conclusion:

For any choice of τ such that $2(c\delta + \eta) < \tau < \Pi - 3(c\delta + \eta)$, the number of clusters computed by the algorithm is equal to the number of peaks of f with probability at least $1 - e^{-\Omega(n)}$.

(the Ω notation hides factors depending on c, δ and the sectional curvature of $\mathbb{X}$)

Estimating the Correct Number of Clusters

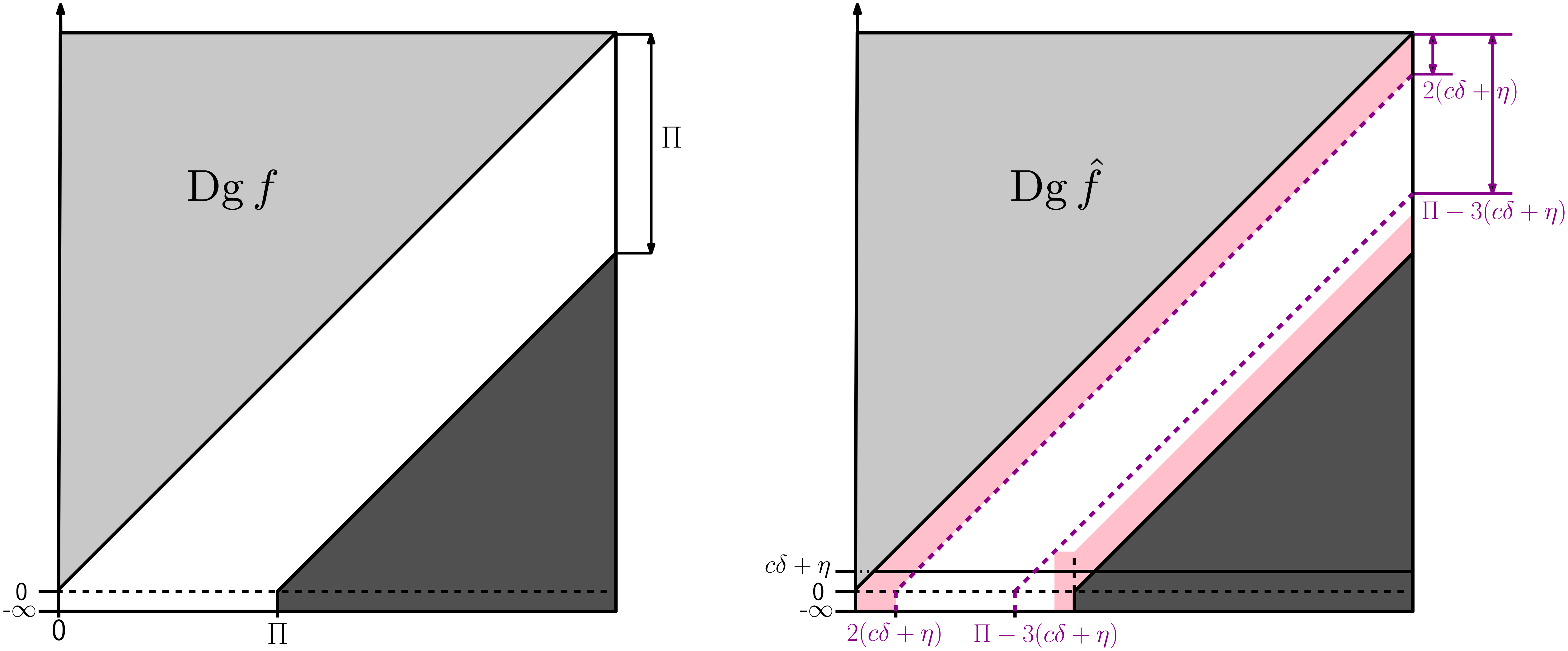


Conclusion:

For any choice of τ such that $2(c\delta+\eta) < \tau < \Pi - 3(c\delta+\eta)$, the number of clusters computed by the algorithm is equal to the number of peaks of f with probability at least $1 - e^{-\Omega(n)}$.

(the Ω notation hides factors depending on c, δ and the sectional curvature of $\mathbb{X}$)

Estimating the Correct Number of Clusters

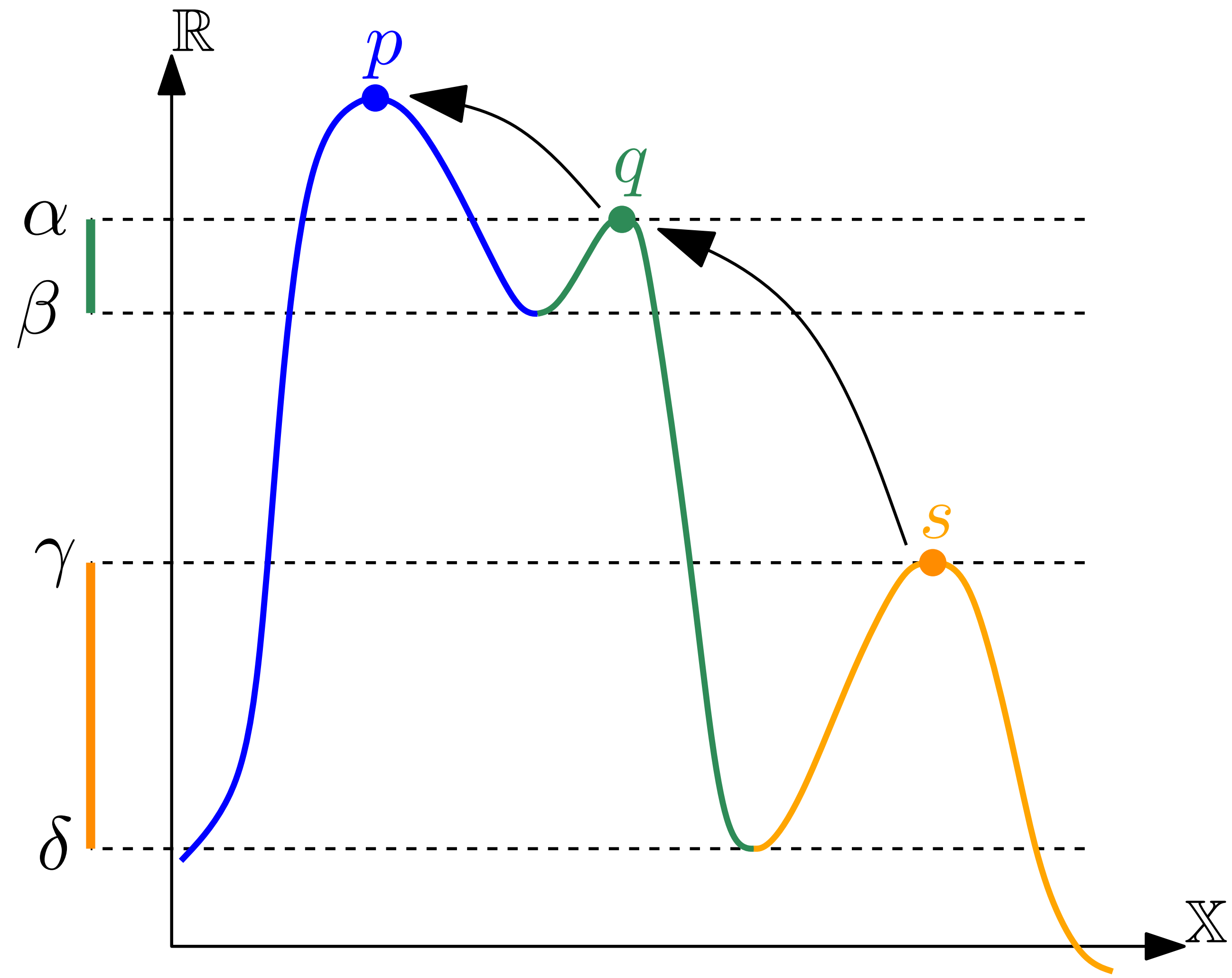


Proof's main ingredient: stability theorem for persistence diagrams

Note: $f, \hat{f}$ are not defined over the same domain

Merging Clusters

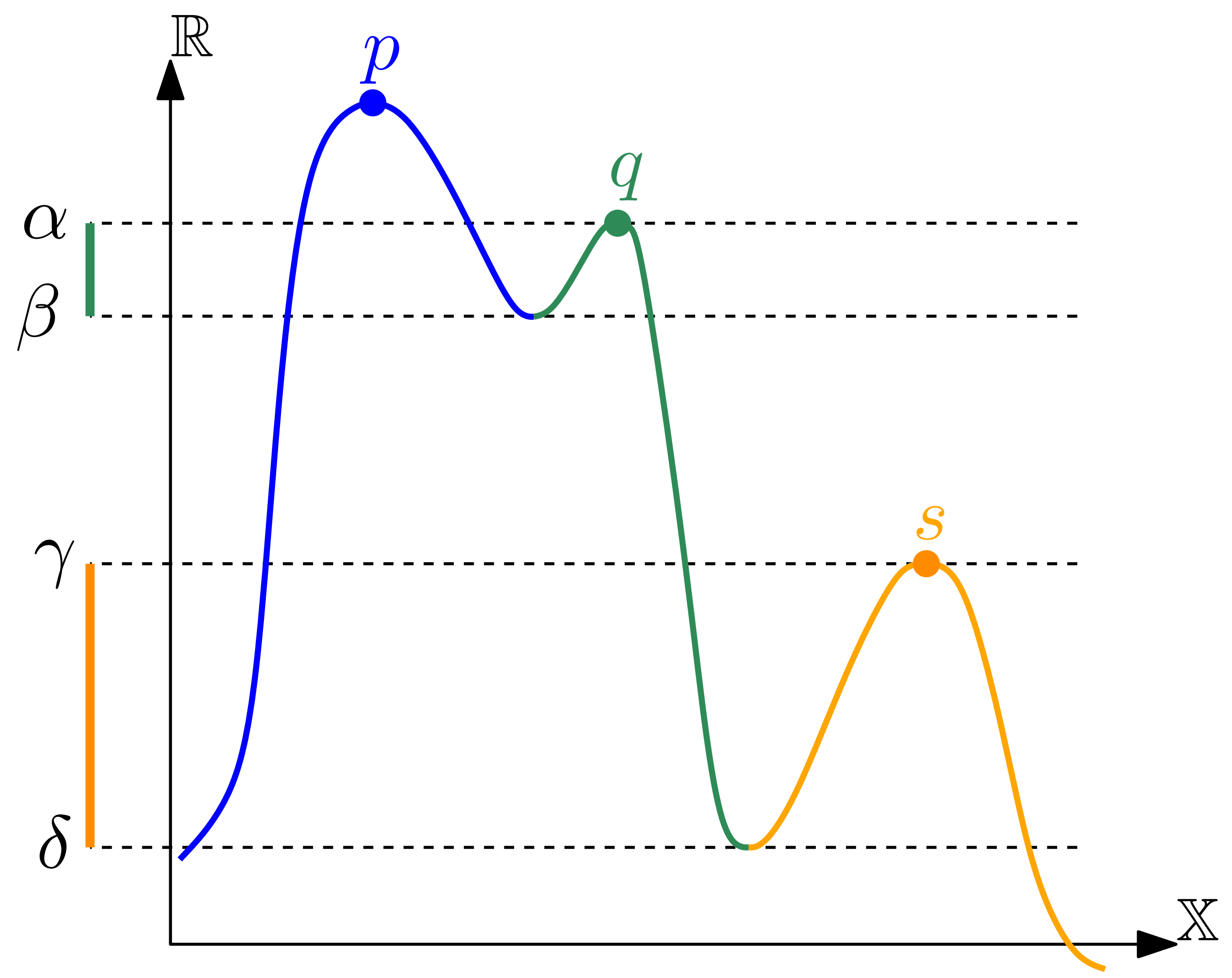
- 0-dimensional persistence builds a hierarchy of the peaks of $\hat{f}$ (merge tree)
- merge clusters according to the hierarchy (merge each cluster into its parent)



Merging Clusters

- 0-dimensional persistence builds a hierarchy of the peaks of $\hat{f}$ (merge tree)
- merge clusters according to the hierarchy (merge each cluster into its parent)
- given a fixed threshold $\tau \geq 0$, only merge those clusters of prominence $< \tau$

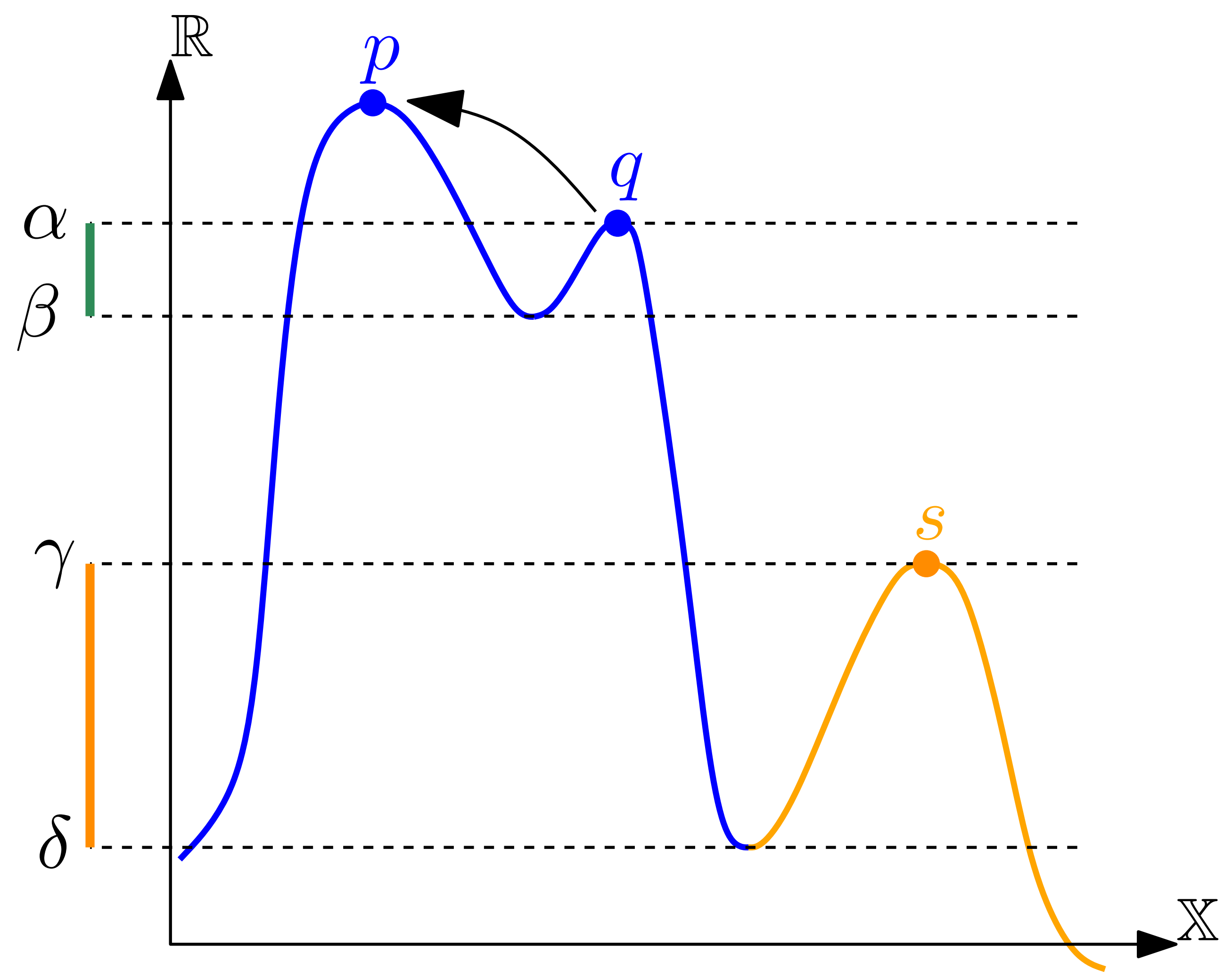
$$0 \leq \tau \leq \alpha - \beta$$



Merging Clusters

- 0-dimensional persistence builds a hierarchy of the peaks of $\hat{f}$ (merge tree)
- merge clusters according to the hierarchy (merge each cluster into its parent)
- given a fixed threshold $\tau \geq 0$, only merge those clusters of prominence $< \tau$

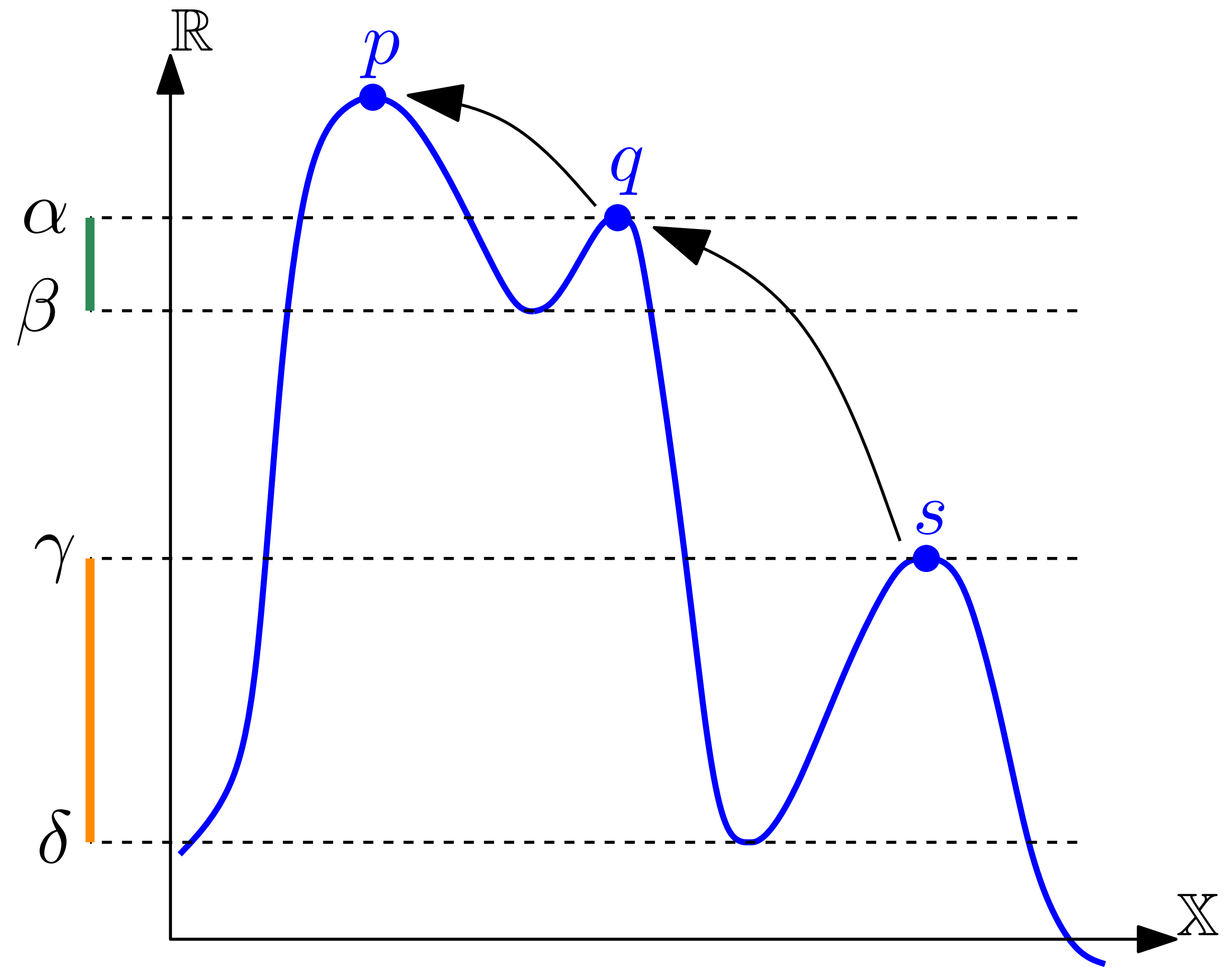
$$\alpha - \beta < \tau \leq \gamma - \delta$$



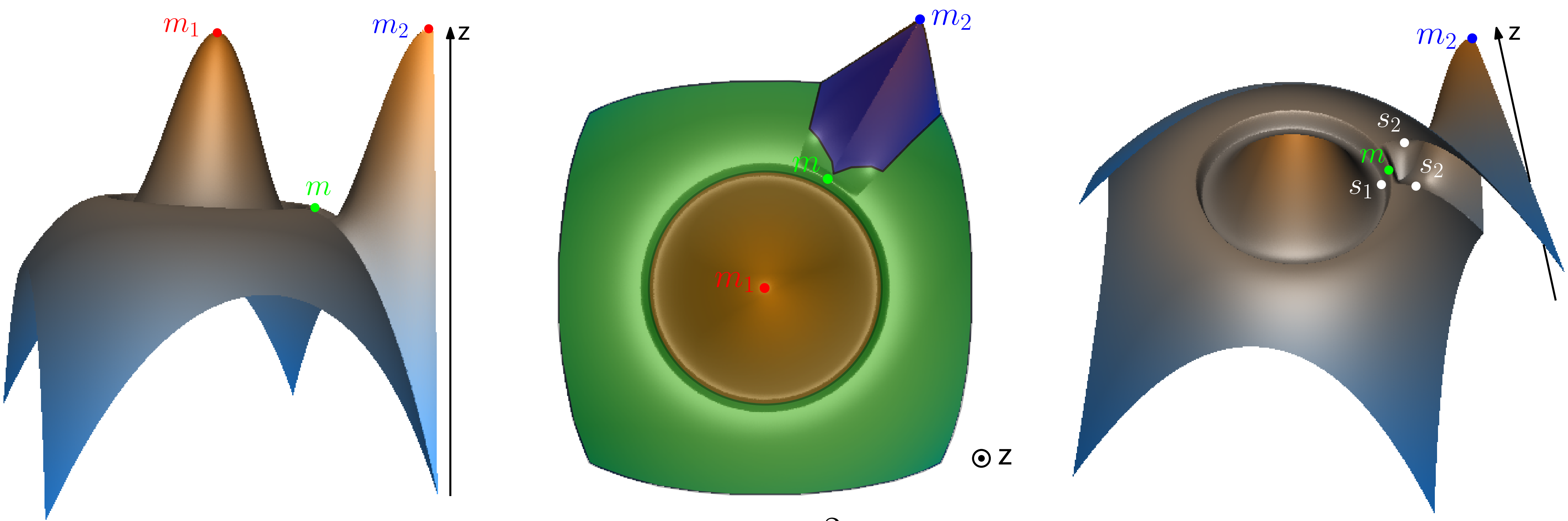
Merging Clusters

- 0-dimensional persistence builds a hierarchy of the peaks of $\hat{f}$ (merge tree)
- merge clusters according to the hierarchy (merge each cluster into its parent)
- given a fixed threshold $\tau \geq 0$, only merge those clusters of prominence $< \tau$

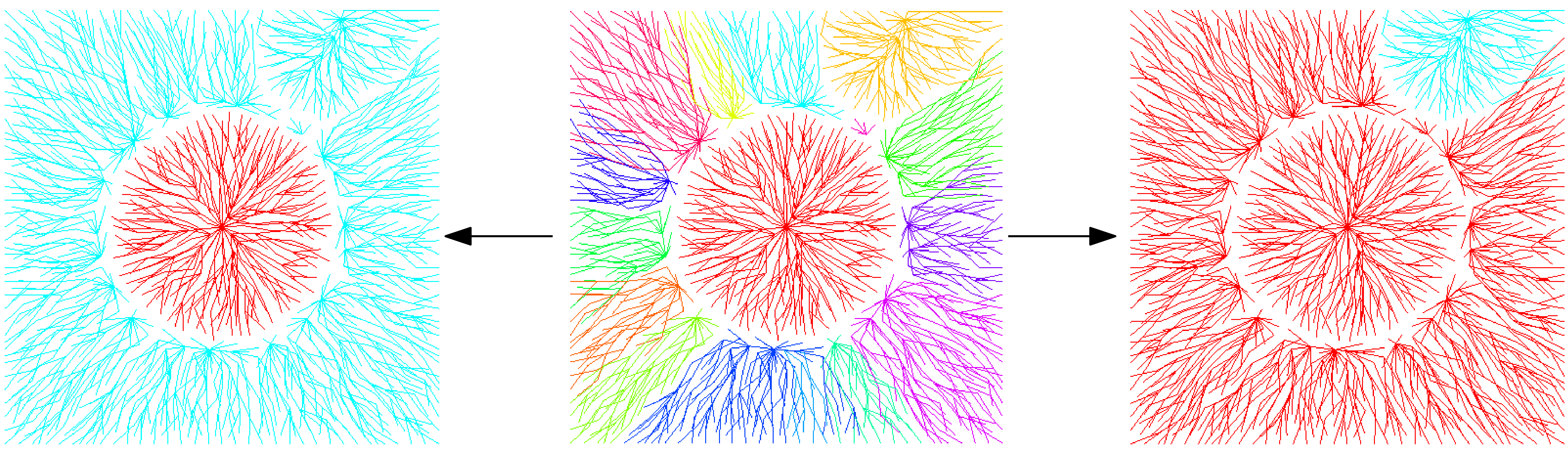
$$\gamma - \delta < \tau \leq +\infty$$



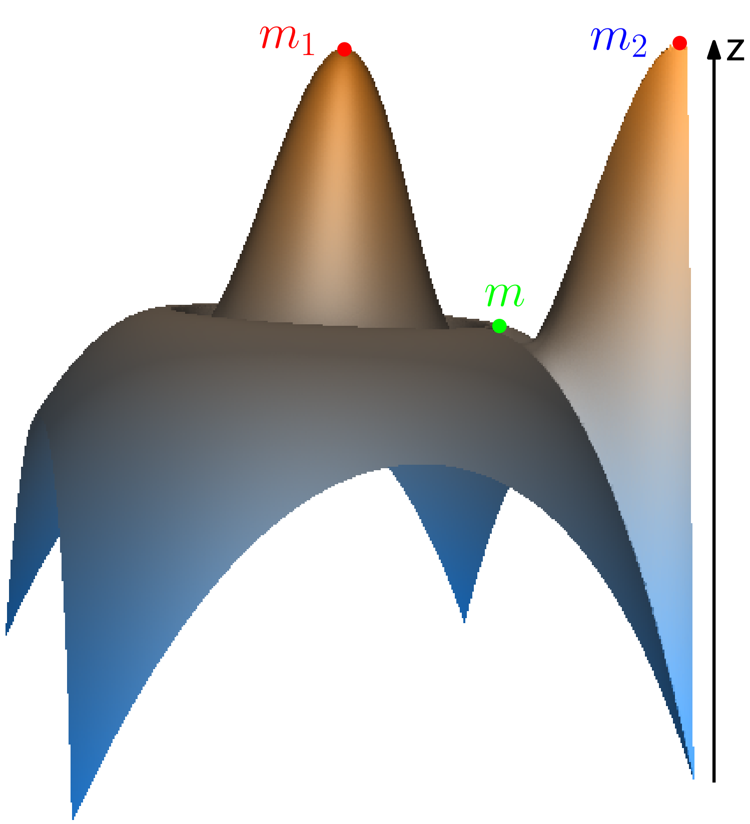
Approximating the Basins of Attraction of f

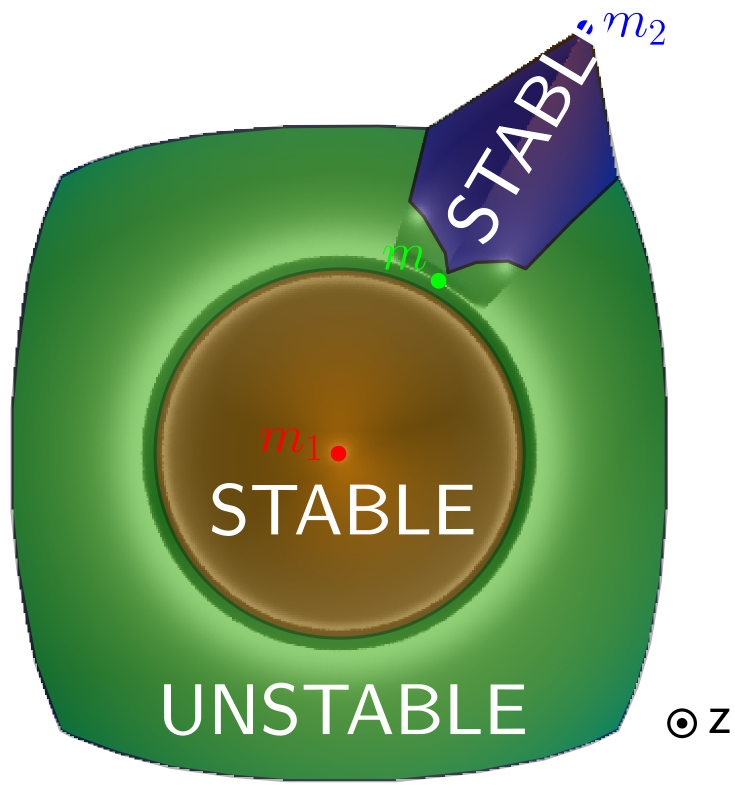


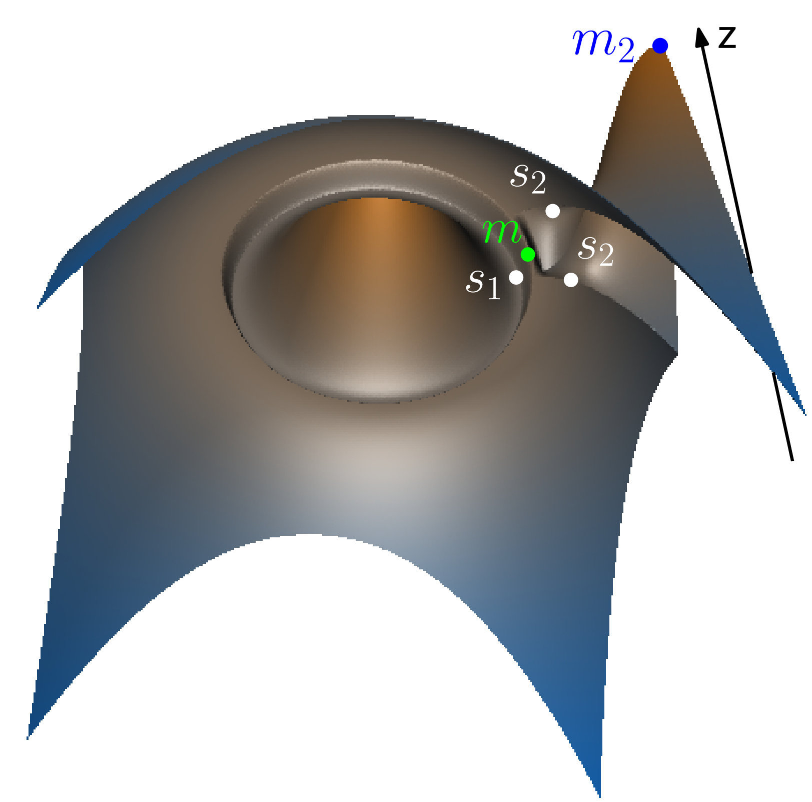
$f : [0,1]^2 \to \mathbb{R}$



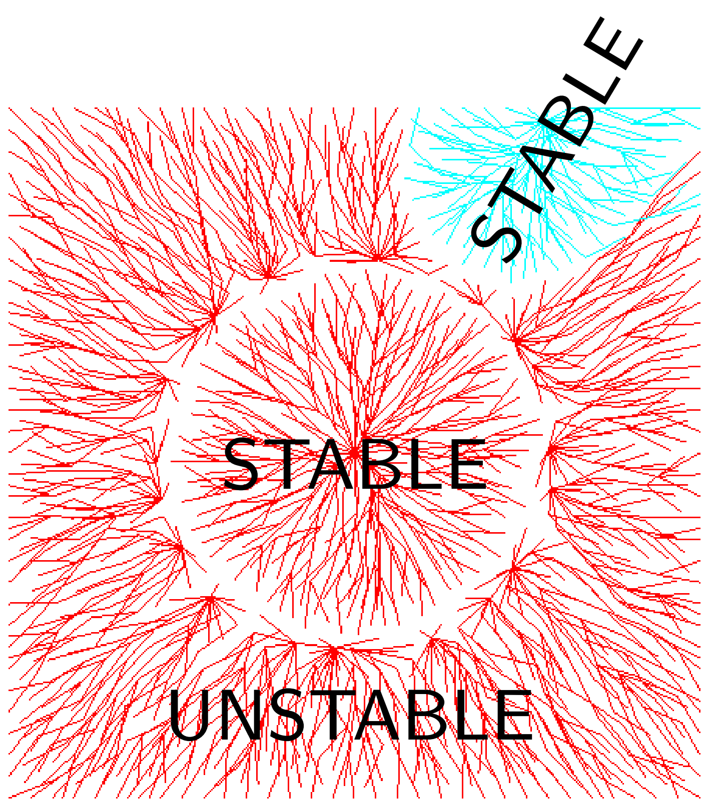
Approximating the Basins of Attraction of f







Partial Approximation Theorem: the cluster associated with a τ-prominent peak in the graph is the *trace* over P of the (merged) basin of attraction of the corresponding peak in the underlying continuous domain, until that basin gets connected to the one of another τ-prominent peak.



Complexity of the Algorithm

Given a neighborhood graph with n vertices (with density values) and m edges:

1. the algorithm sorts the vertices by decreasing density values,

2. the algorithm makes a single pass through the vertex set, creating the spanning forest and merging clusters on the fly using a union-find data structure.

Complexity of the Algorithm

Given a neighborhood graph with n vertices (with density values) and m edges:

1. the algorithm sorts the vertices by decreasing density values,

2. the algorithm makes a single pass through the vertex set, creating the spanning forest and merging clusters on the fly using a union-find data structure.

$\rightarrow$ Running time: $O(n \log n + (n+m)\alpha(n))$

$\rightarrow$ Space complexity: $O(n+m)$

$\rightarrow$ Main memory usage: $O(n)$

Pseudo-code:

Input: simple graph G with n vertices, n-dimensional vector $\tilde{f}$, real parameter $\tau \geq 0$.

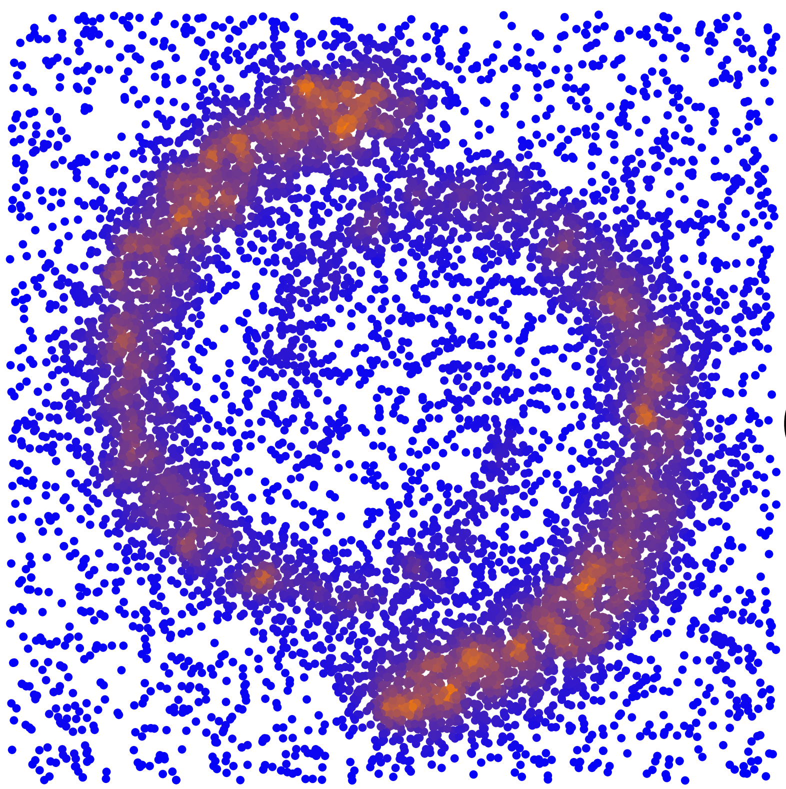
Sort the vertex indices $\{1, 2, \cdots, n\}$ so that $\tilde{f}(1) \geq \tilde{f}(2) \geq \cdots \geq \tilde{f}(n)$;
Initialize a union-find data structure $\mathcal{U}$ and two vectors g, r of size n;

```
for i = 1 to n do
    Let N be the set of neighbors of i in G that have indices lower than i;
    if N = ∅  // vertex i is a peak of f̃ within G
        Create a new entry e in U and attach vertex i to it;
        r(e) ← i  // r(e) stores the root vertex associated with the entry e
    else  // vertex i is not a peak of f̃ within G
        g(i) ← argmax_{j∈N} f̃(j)  // g(i) stores the approximate gradient at vertex i
        e_i ← U.find(g(i));
        Attach vertex i to the entry e_i;
        for j ∈ N do
            e ← U.find(j);
            if e ≠ e_i and min{f̃(r(e)), f̃(r(e_i))} < f̃(i) + τ
                U.union(e, e_i);
                r(e ∪ e_i) ← argmax_{{r(e), r(e_i)}} f̃;
                e_i ← e ∪ e_i;
```

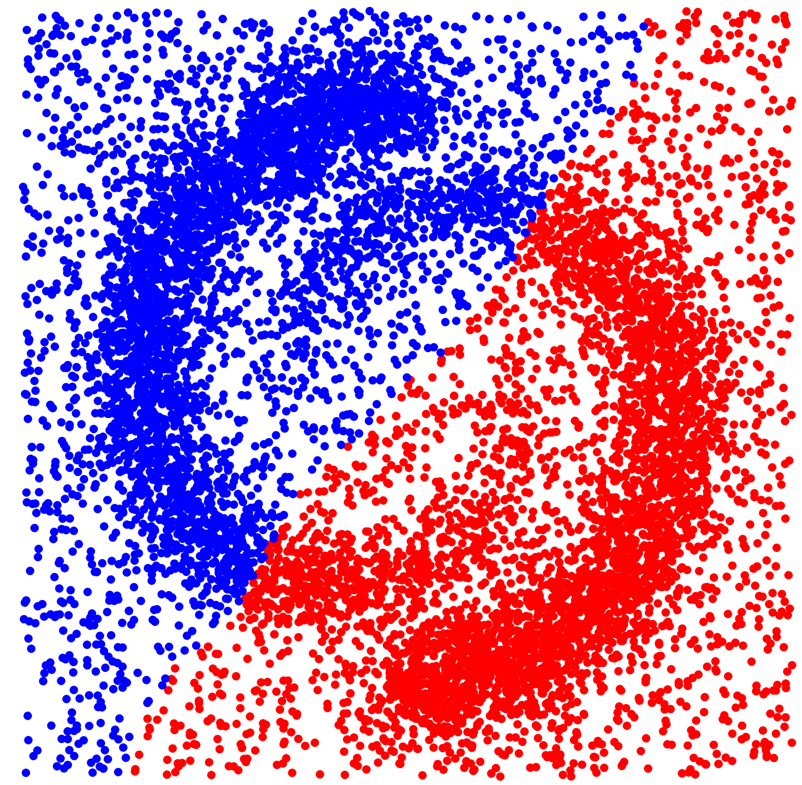
Output: the collection of entries e of $\mathcal{U}$ such that $\tilde{f}(r(e)) \geq \tau$.

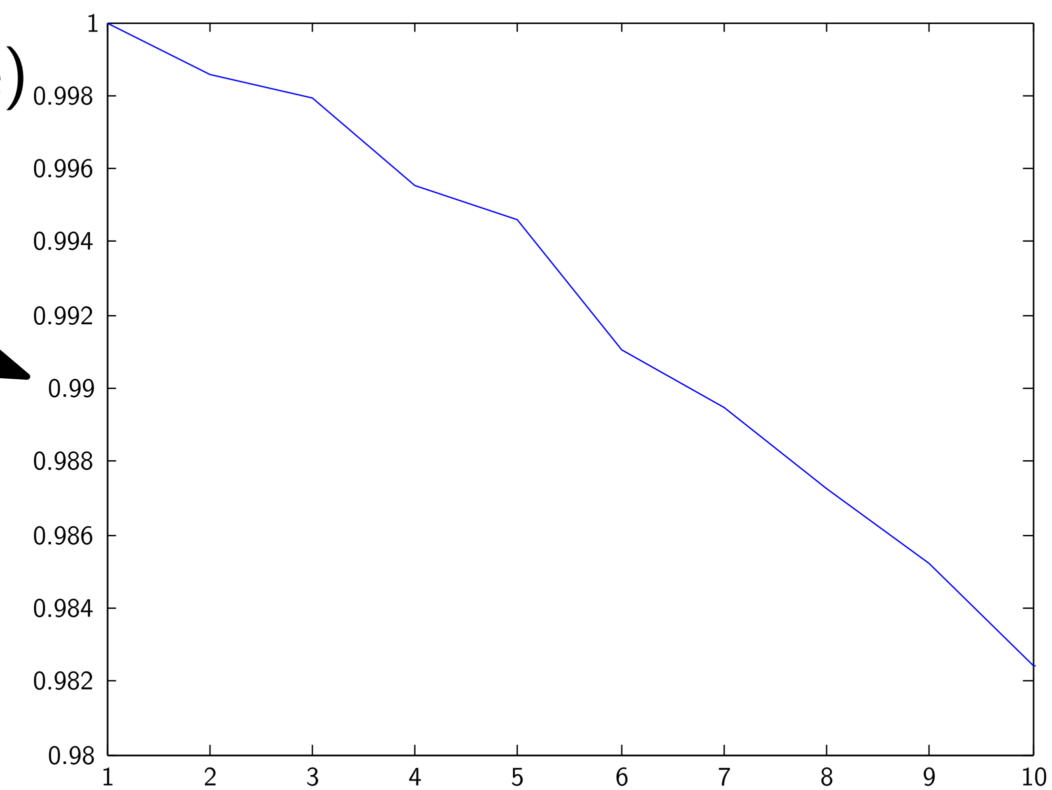
Experimental Results

Synthetic Data



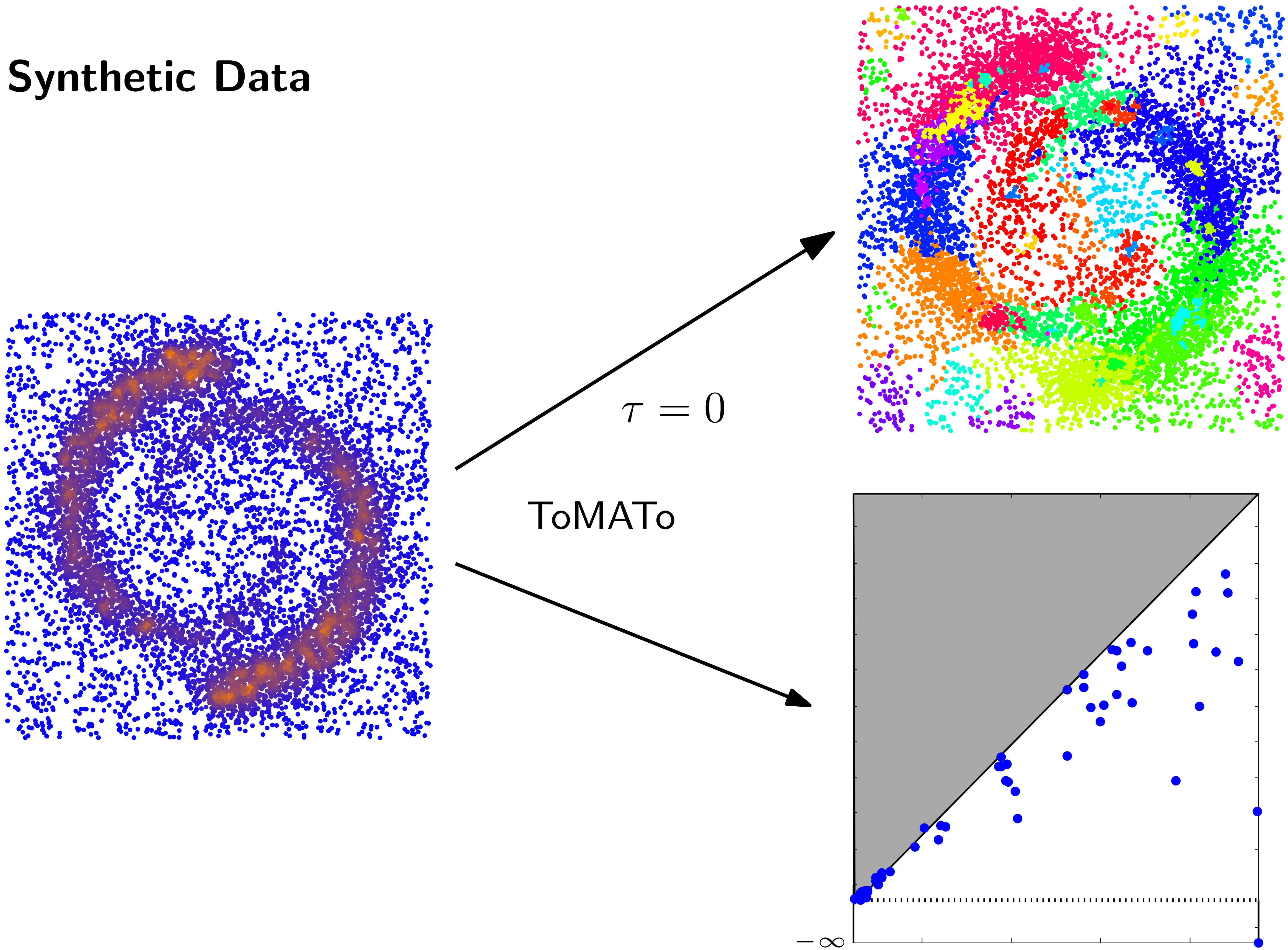
Spectral clustering
(k-means in eigenspace)





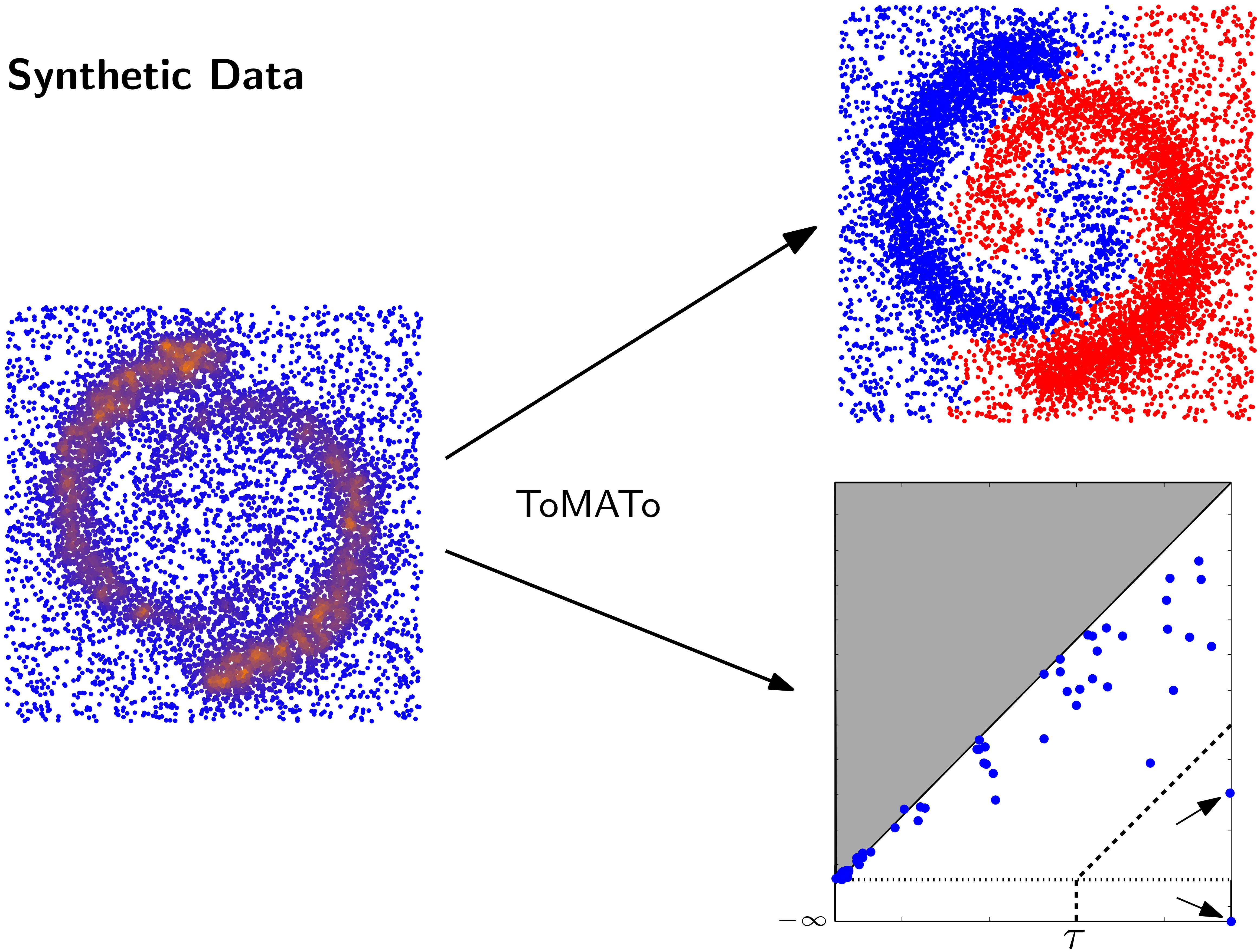
Experimental Results

Synthetic Data



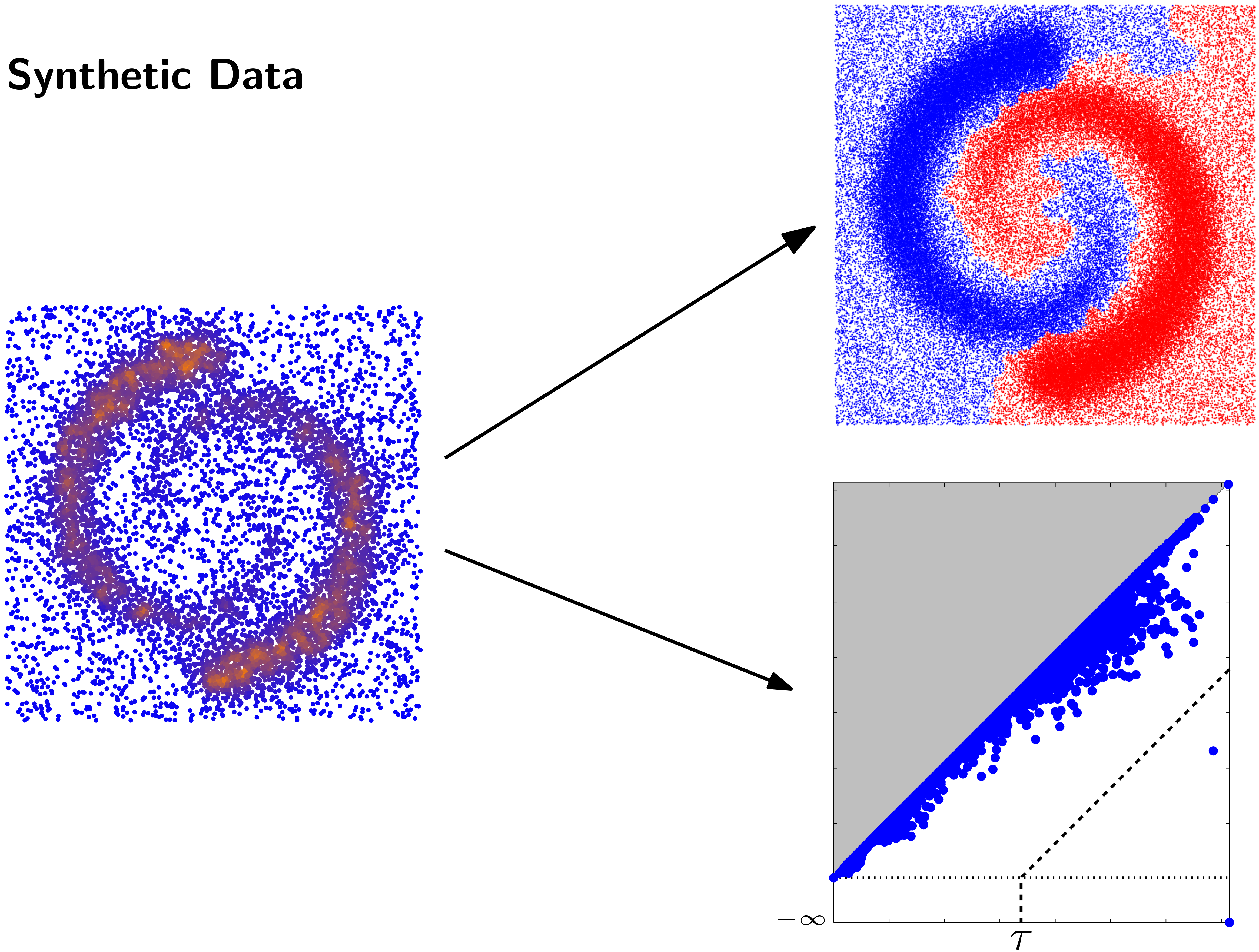
Experimental Results

Synthetic Data



Experimental Results

Synthetic Data

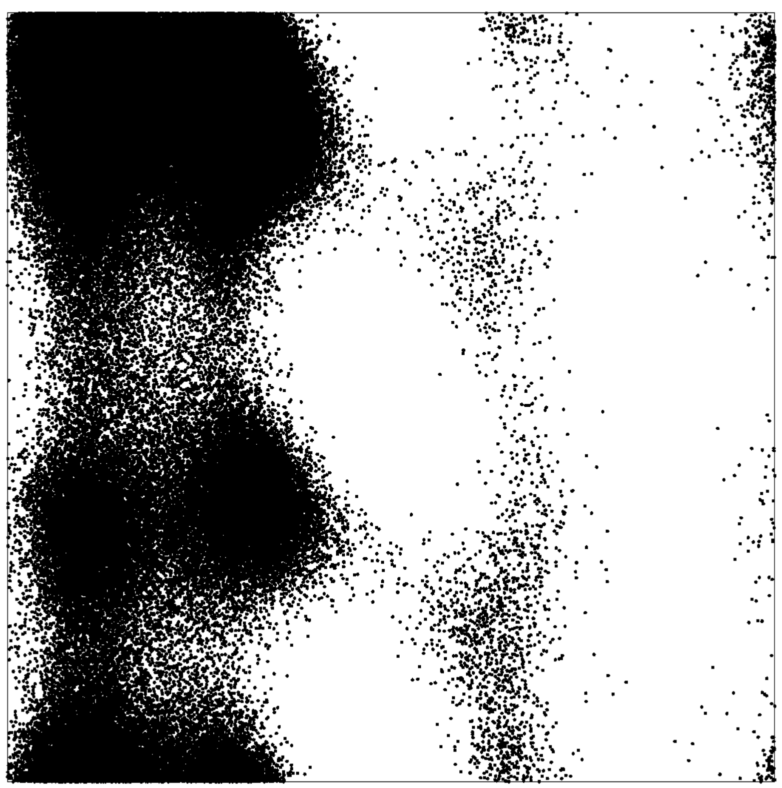


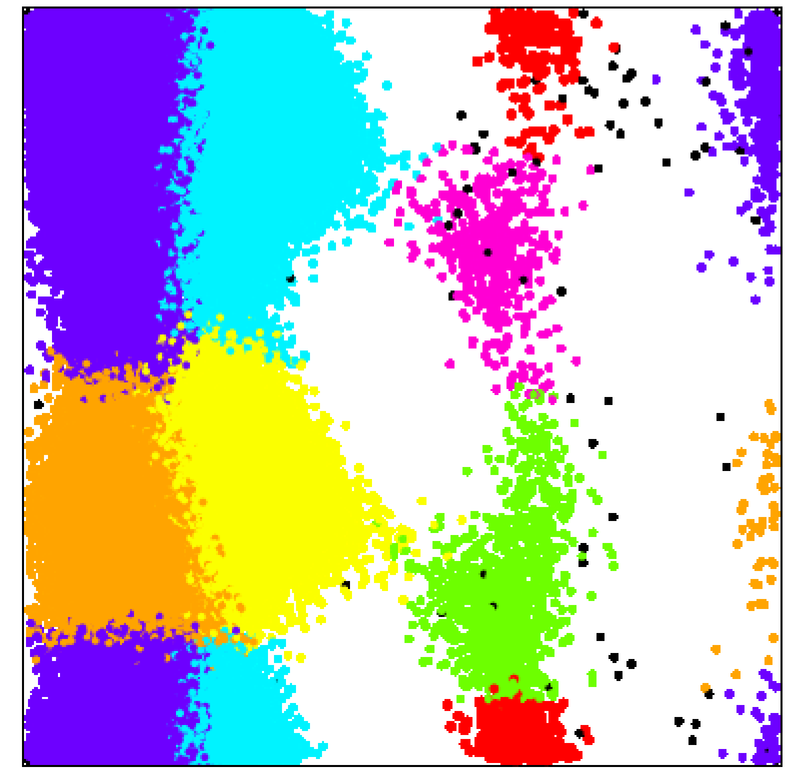
Experimental Results

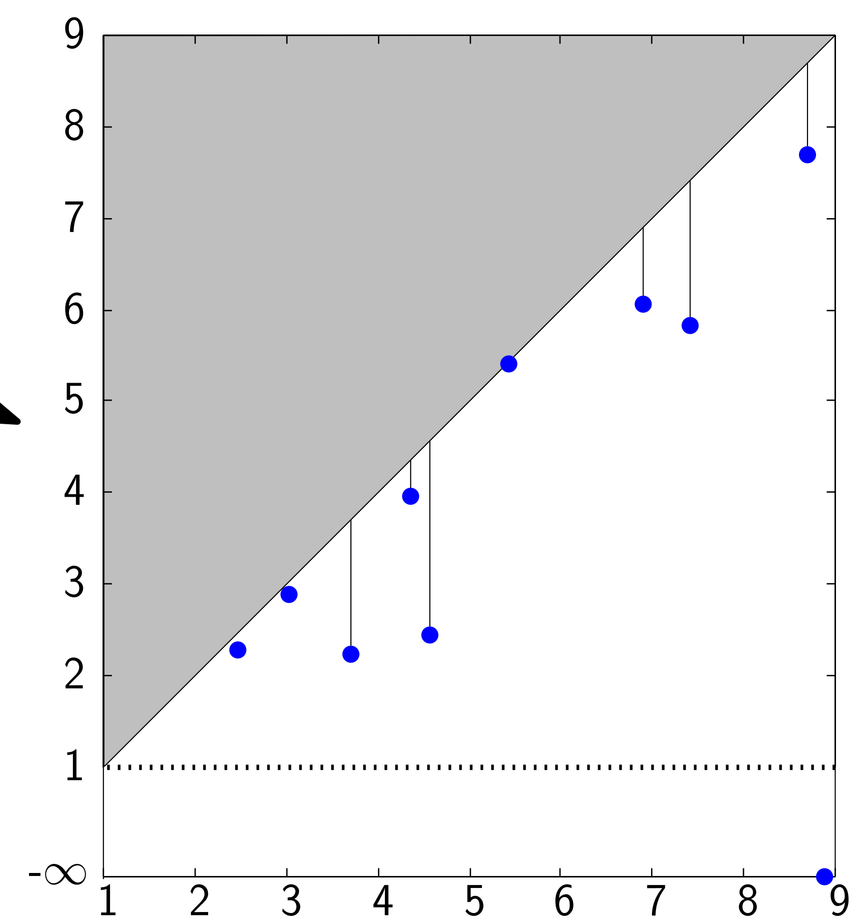
Biological Data

Alanine-Dipeptide conformations ($\mathbb{R}^{21}$)

RMSD distance (non-Euclidean)







Common belief: 6 metastable states

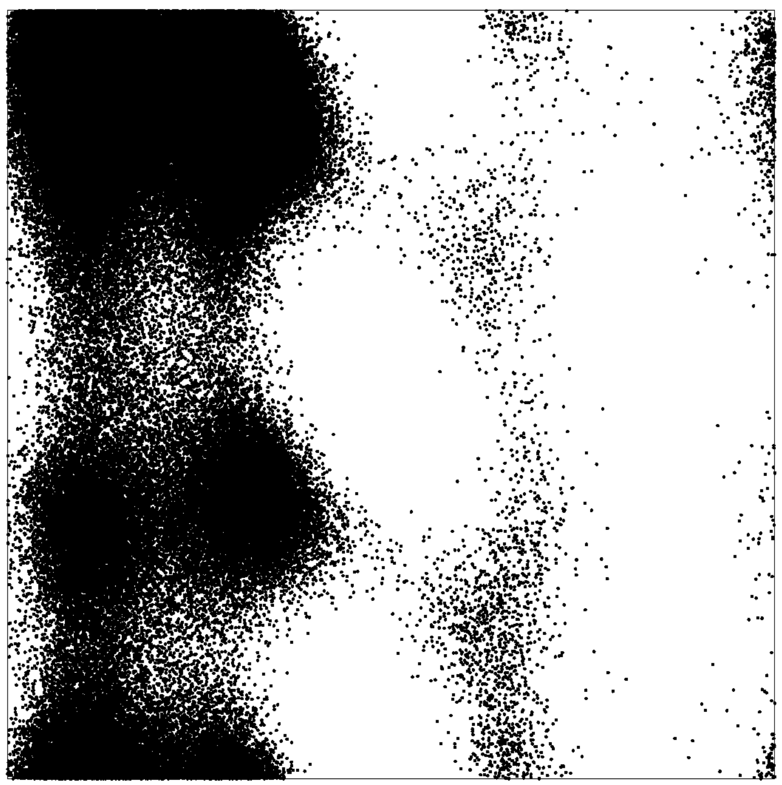
PD shows anywhere between 4 and 7 clusters

Experimental Results

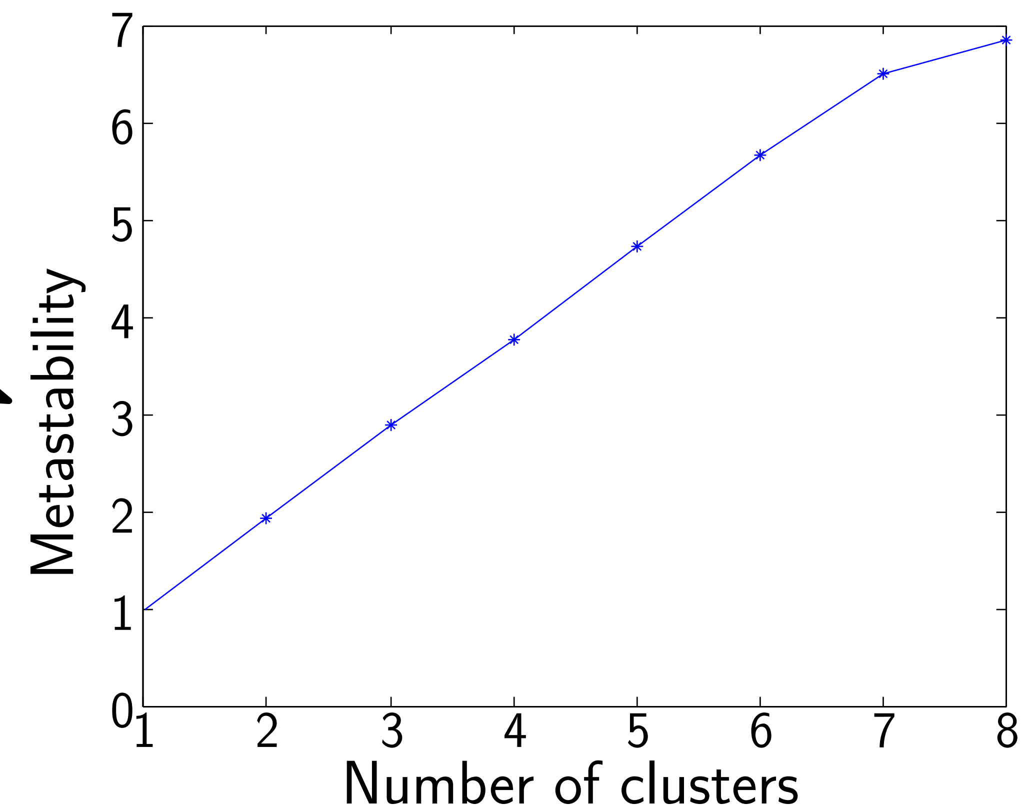
Biological Data

Alanine-Dipeptide conformations ($\mathbb{R}^{21}$)

RMSD distance (non-Euclidean)



Rank	Prominence	Metastability
1	$+\infty$	0.99982
2	3827	1.91865
3	1334	2.8813
4	557	3.76217
5	85	4.73838
6	32	5.65553
7	26	6.50757
8	7.2	6.8193
9	3.0	-
10	2.2	-



Common belief: 6 metastable states

PD shows anywhere between 4 and 7 clusters

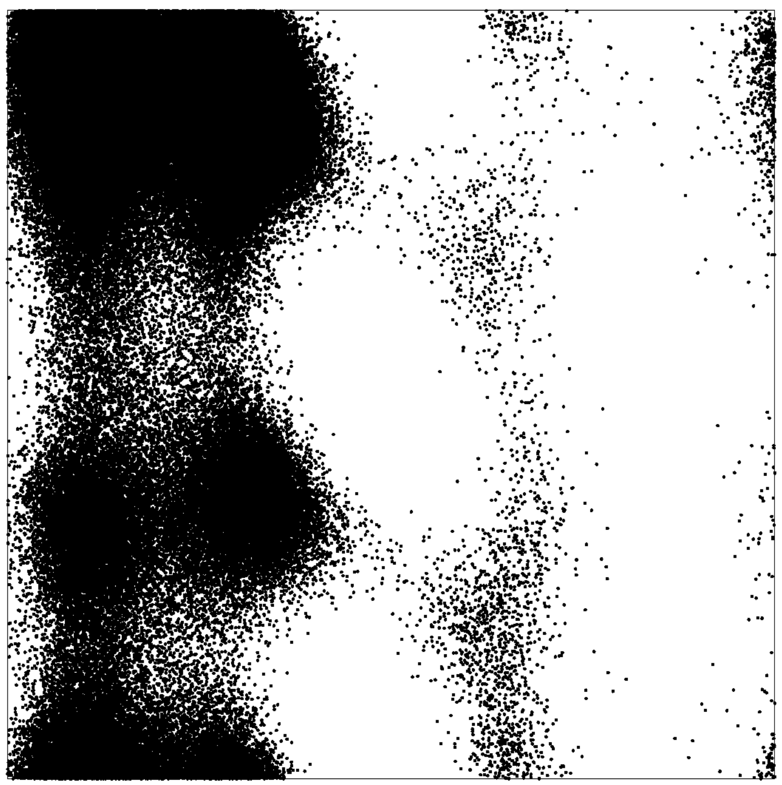
Measures of metastability confirm this insight

Experimental Results

Biological Data

Alanine-Dipeptide conformations ($\mathbb{R}^{21}$)

RMSD distance (non-Euclidean)



Note: Spectral Clustering takes a week of tweaking, while ToMATo runs out-of-the-box in a few minutes

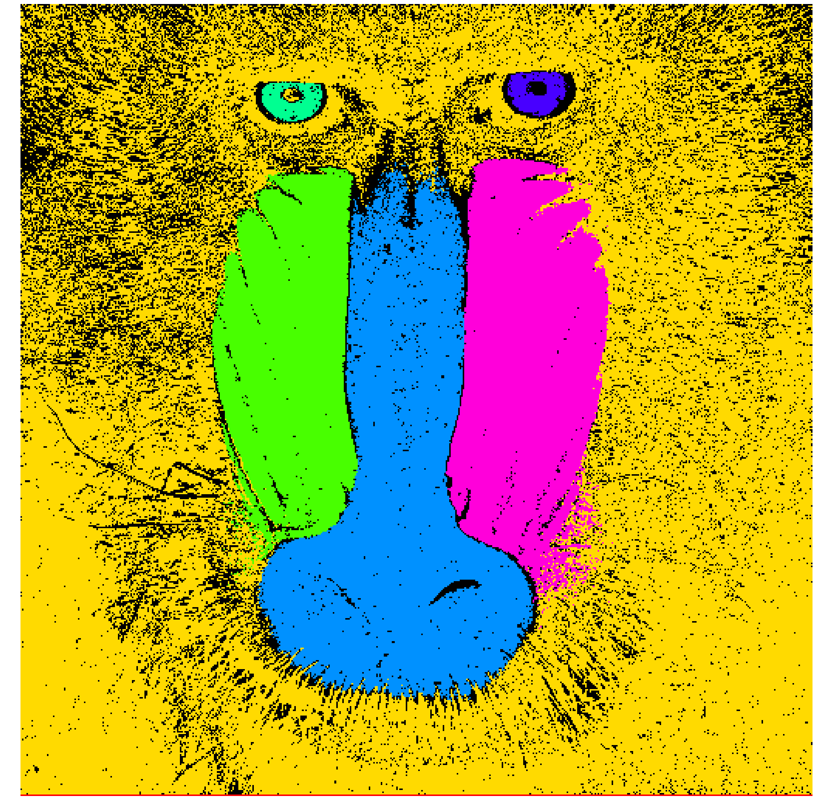
• Y. Yao, J. Sun, X. Huang, G. Bowman, G. Singh, M. Lesnick, L. Guibas, V. Pande, G. Carlsson, Topological methods for exploring low-density states in biomolecular folding pathways, *The Journal of Chemical Physics*, 2009.

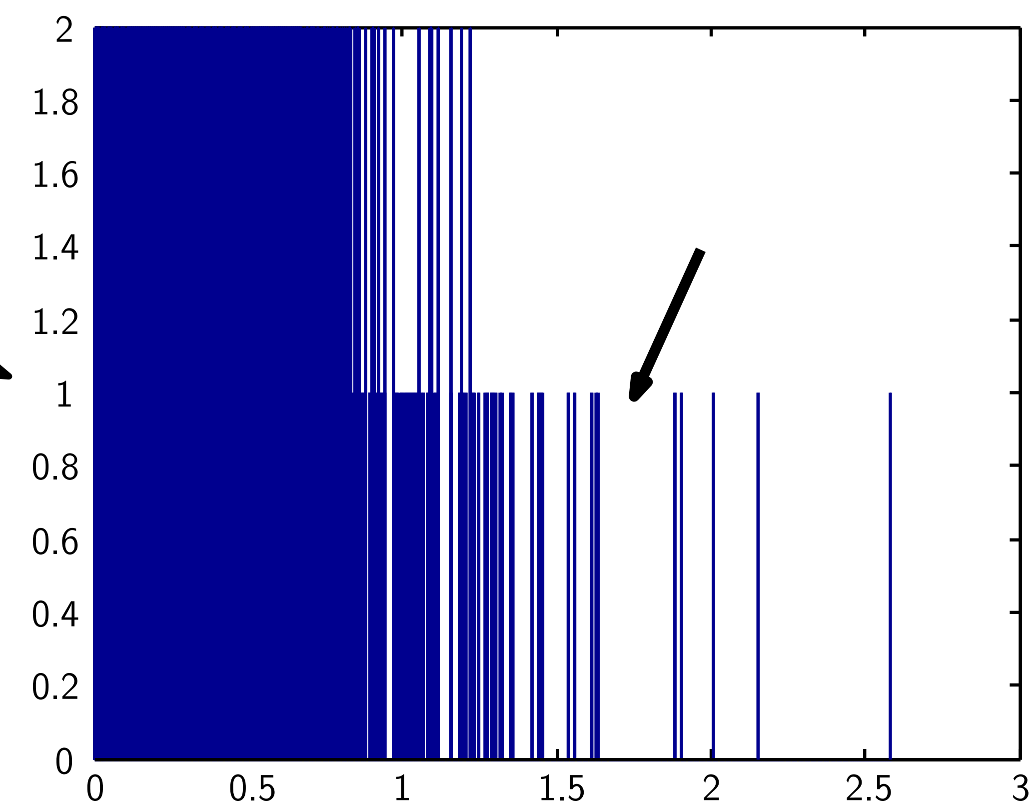
Experimental Results

Image Segmentation

Density is estimated in 3D color space (Luv)

Neighborhood graph is built in image domain





Distribution of prominences does not usually show a clear unique gap

Still, relationship between choice of τ and number of obtained clusters remains explicit

Recap'

ToMATo:

1. graph-based mode-seeking algorithm of [KNF'76]
2. single-pass cluster merging phase guided by persistence

Competitors:

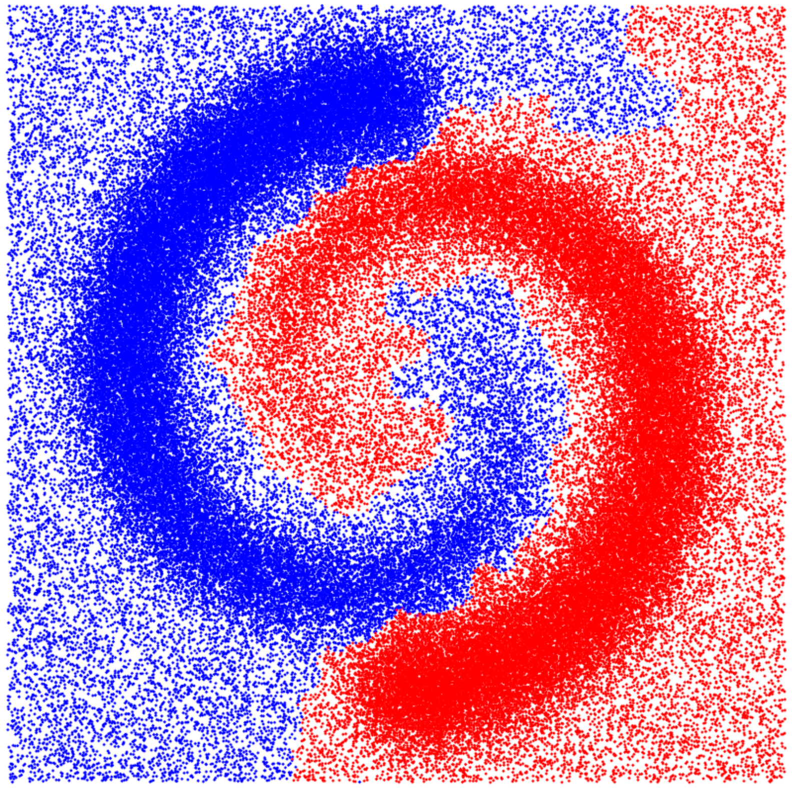
1. Mean-Shift and its variants (smoothing a priori)
2. ...

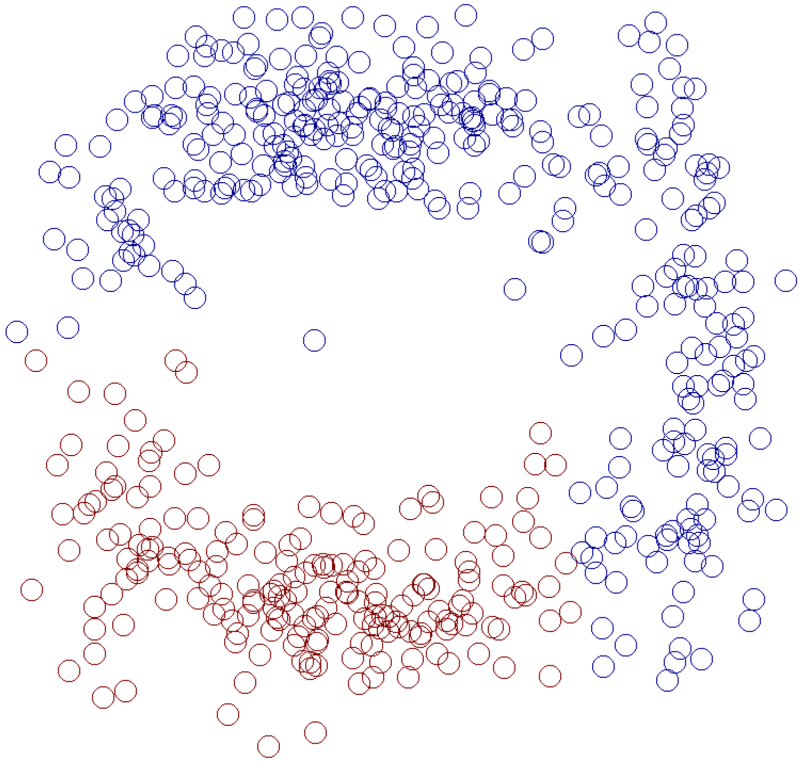
Recap'

- Highly generic
  - applicable in arbitrary metric spaces
  - agnostic to the choice of neighborhood graph and density estimator
- Easy to tune
  - mostly two parameters: neighborhood size, persistence threshold τ
  - PD provides insight into the correct number of clusters
- Comes with theoretical guarantees
  - number of obtained clusters versus number of prominent peaks
  - partial approximation of the basins of attraction of the peaks
- Efficient and practical
  - near linear runtime, linear main memory usage
  - can handle data sets with hundreds of thousands of points in practice

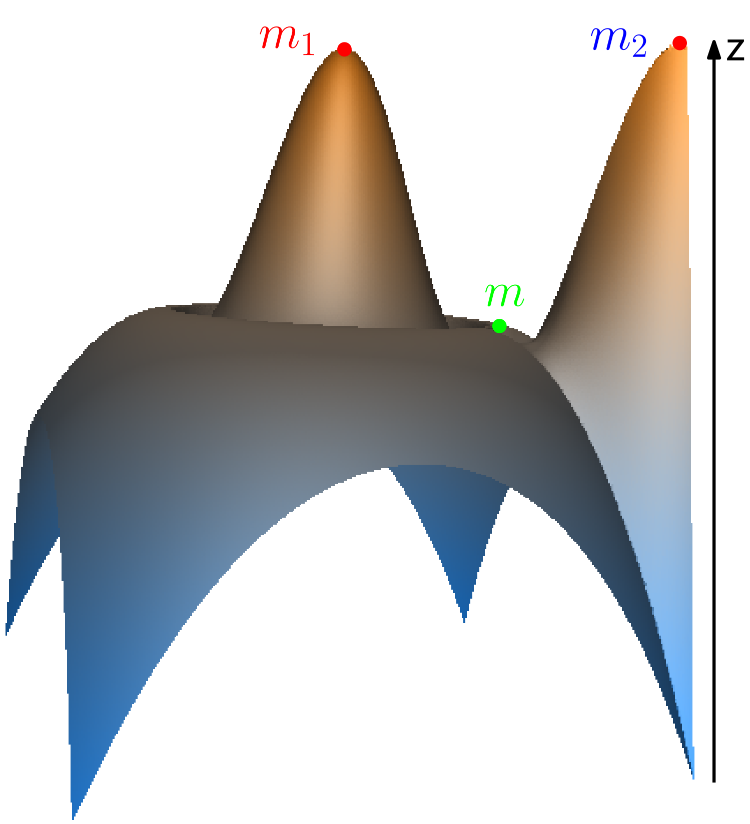
Recap'

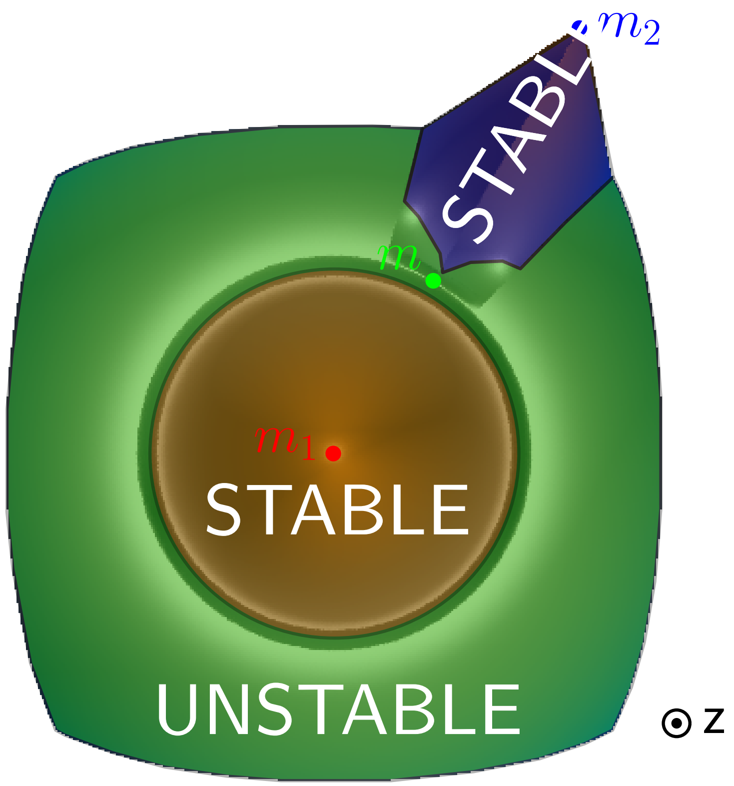
Q Can we devise soft variants?

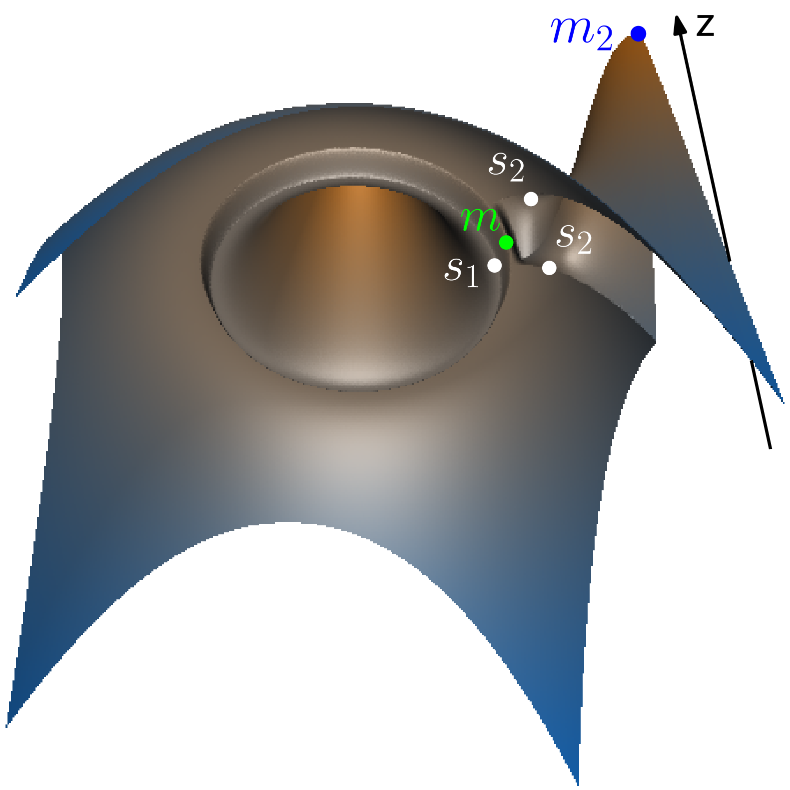




First idea: add randomness to the estimator

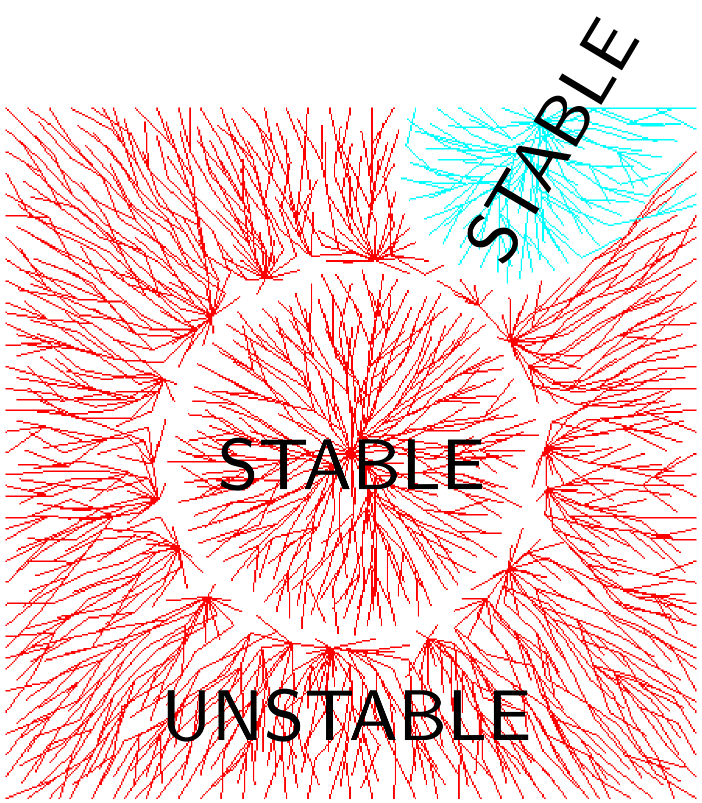




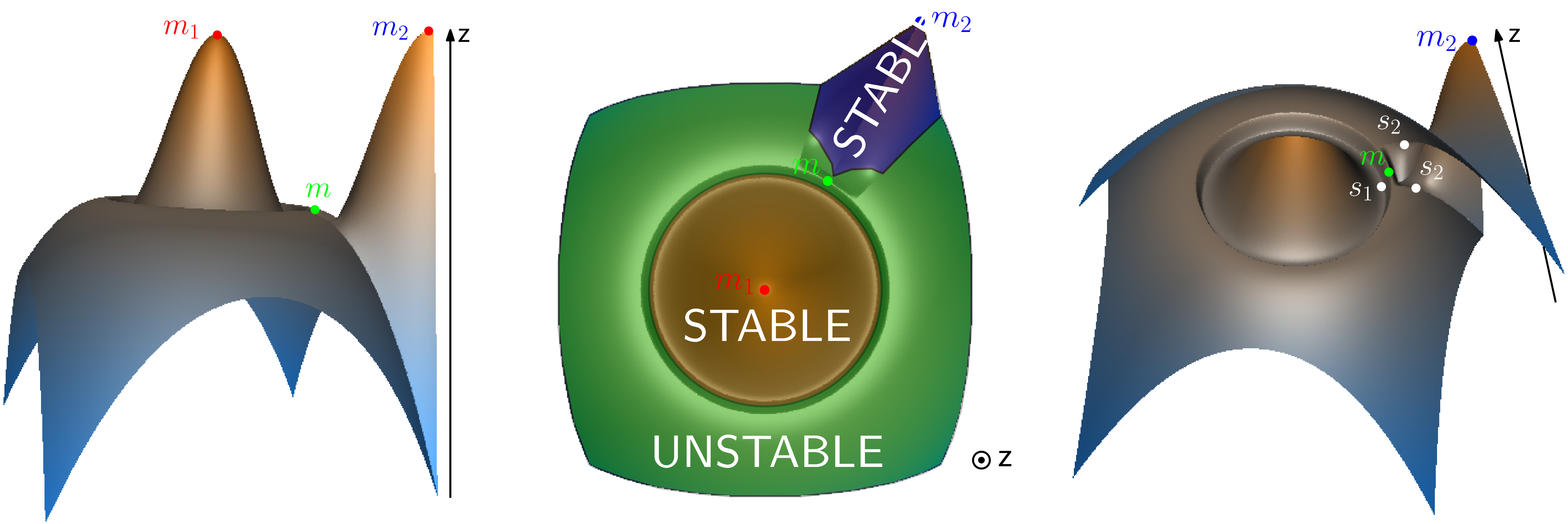


→ soft clustering variant [Skraba et al. '10]:

- rerun the algorithm with randomly perturbed function values
- identify τ-prominent clusters across different runs
- assign points to clusters with probabilities depending on the outcomes of the runs



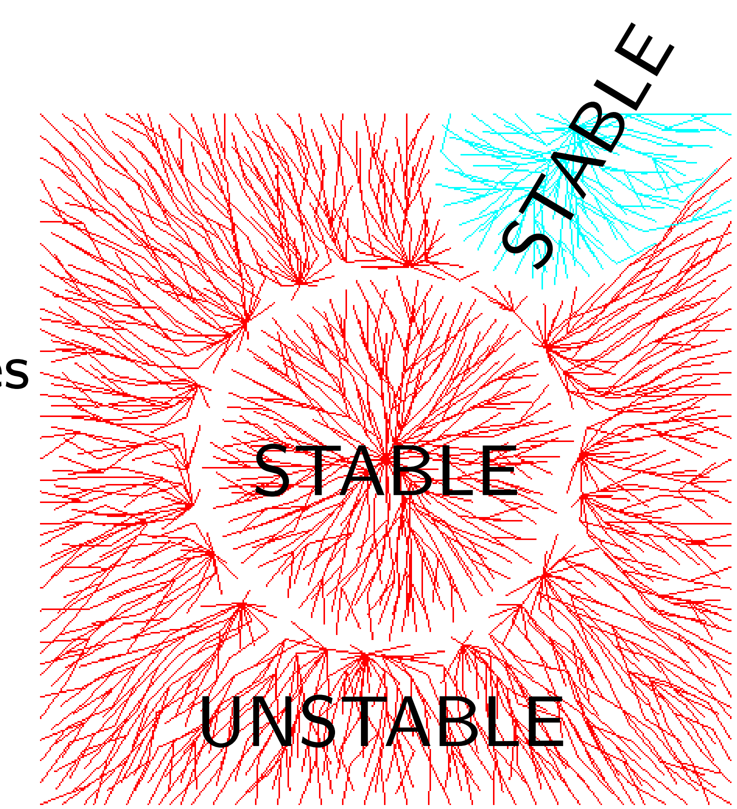
First idea: add randomness to the estimator



Pb: What is the corresponding continuous process?

$\rightarrow$ soft clustering variant [Skraba et al. '10]:

- rerun the algorithm with randomly perturbed function values
- identify τ-prominent clusters across different runs
- assign points to clusters with probabilities depending on the outcomes of the runs



Second idea: add randomness to gradient ascent

Given a density $f : \mathbb{R}^d \to \mathbb{R}$ and a point $x \in \mathbb{R}^d$ s.t. $f(x) > 0$, consider the SDE:

$$
\begin{aligned}
&dX_t = \nabla \log f(X_t) + \sqrt{\beta} I_d dW_t \\
&X_0 = x
\end{aligned}
$$

Second idea: add randomness to gradient ascent

Given a density $f : \mathbb{R}^d \to \mathbb{R}$ and a point $x \in \mathbb{R}^d$ s.t. $f(x) > 0$, consider the SDE:

$$
\begin{aligned}
&dX_t = \nabla \log f(X_t) + \sqrt{\beta} I_d dW_t \\
&X_0 = x
\end{aligned}
$$

gradient term   temperature   isotropic diffusion term

$\beta = 0 \to$ pure mode seeking

$\beta \to +\infty \to$ pure diffusion

Second idea: add randomness to gradient ascent

Given a density $f : \mathbb{R}^d \to \mathbb{R}$ and a point $x \in \mathbb{R}^d$ s.t. $f(x) > 0$, consider the SDE:

$$\begin{aligned} dX_t &= \nabla \log f(X_t) + \sqrt{\beta} I_d dW_t \\ X_0 &= x \end{aligned}$$

gradient term temperature isotropic diffusion term

$\beta = 0 \to$ pure mode seeking

$\beta \to +\infty \to$ pure diffusion

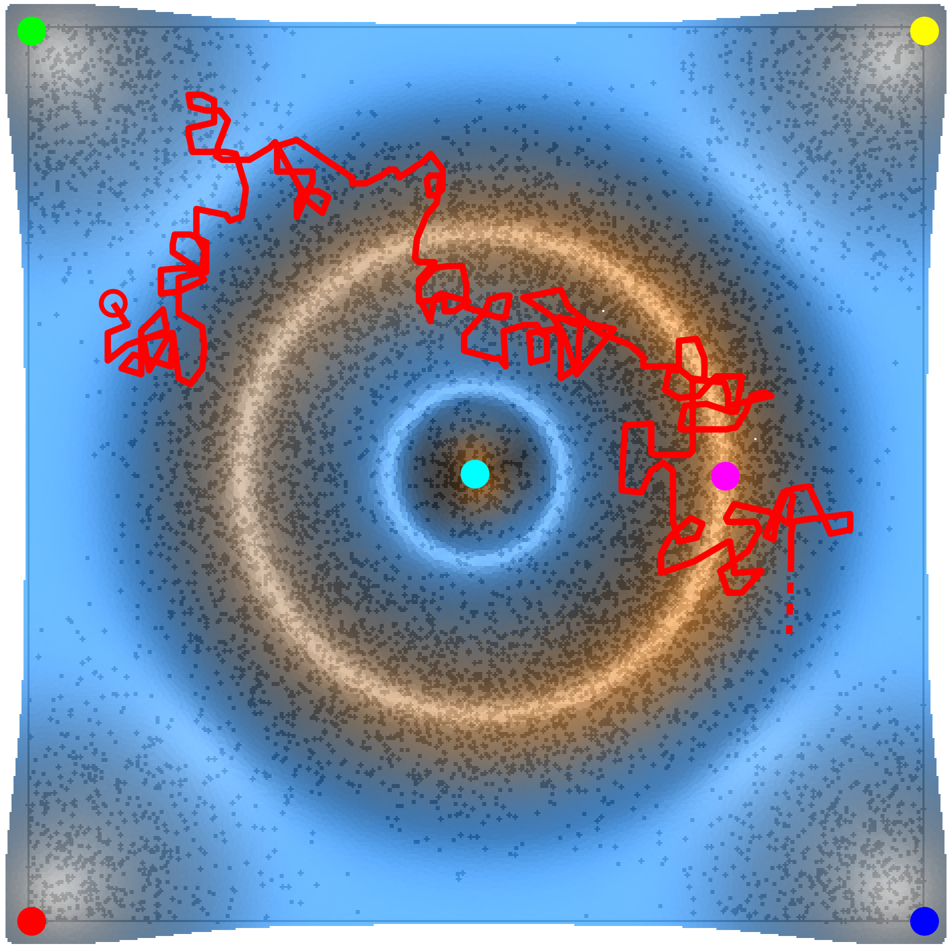
Our continuous process is the solution of this equation (assuming well-posedness)

Second idea: add randomness to gradient ascent

Given a density $f : \mathbb{R}^d \to \mathbb{R}$ and a point $x \in \mathbb{R}^d$ s.t. $f(x) > 0$, consider the SDE:

$$\begin{aligned} dX_t &= \nabla \log f(X_t) + \sqrt{\beta} I_d dW_t \\ X_0 &= x \end{aligned}$$

Pb: probability to reach a given peak of f (or any given point of $\mathbb{R}^d$) is 0



Second idea: add randomness to gradient ascent

Given a density $f : \mathbb{R}^d \to \mathbb{R}$ and a point $x \in \mathbb{R}^d$ s.t. $f(x) > 0$, consider the SDE:

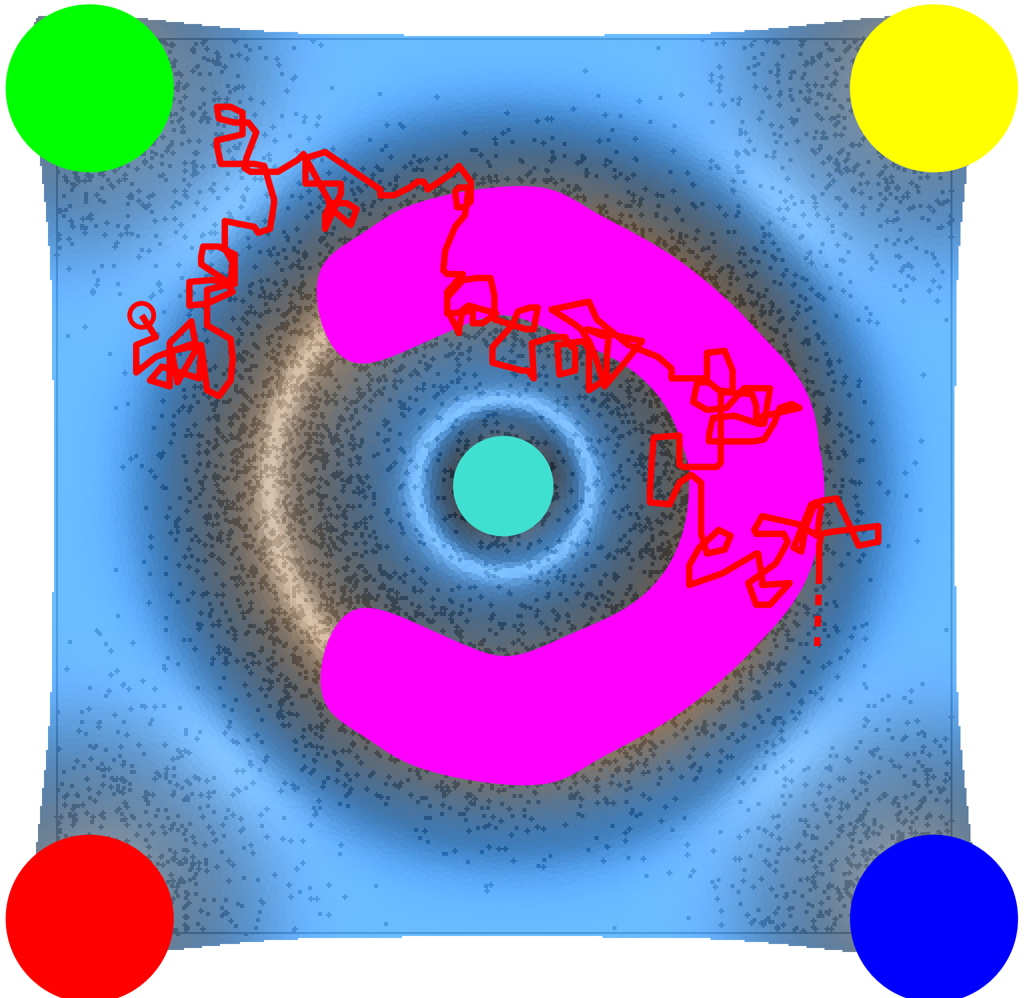
$$\begin{aligned} &dX_t = \nabla \log f(X_t) + \sqrt{\beta} I_d dW_t \\ &X_0 = x \end{aligned}$$

Pb: probability to reach a given peak of f (or any given point of $\mathbb{R}^d$) is 0

$\to$ solution: replace peaks by *cluster cores*:

- belong surely to the basin of attraction of a unique peak of f

- $\mathbb{P}(X_t$ eventually reaches some $C_i) > 0$



Second idea: add randomness to gradient ascent

Given a density $f : \mathbb{R}^d \to \mathbb{R}$ and a point $x \in \mathbb{R}^d$ s.t. $f(x) > 0$, consider the SDE:

$$\begin{aligned} dX_t &= \nabla \log f(X_t) + \sqrt{\beta} I_d dW_t \\ X_0 &= x \end{aligned}$$

Pb: probability to reach a given peak of f (or any given point of $\mathbb{R}^d$) is 0

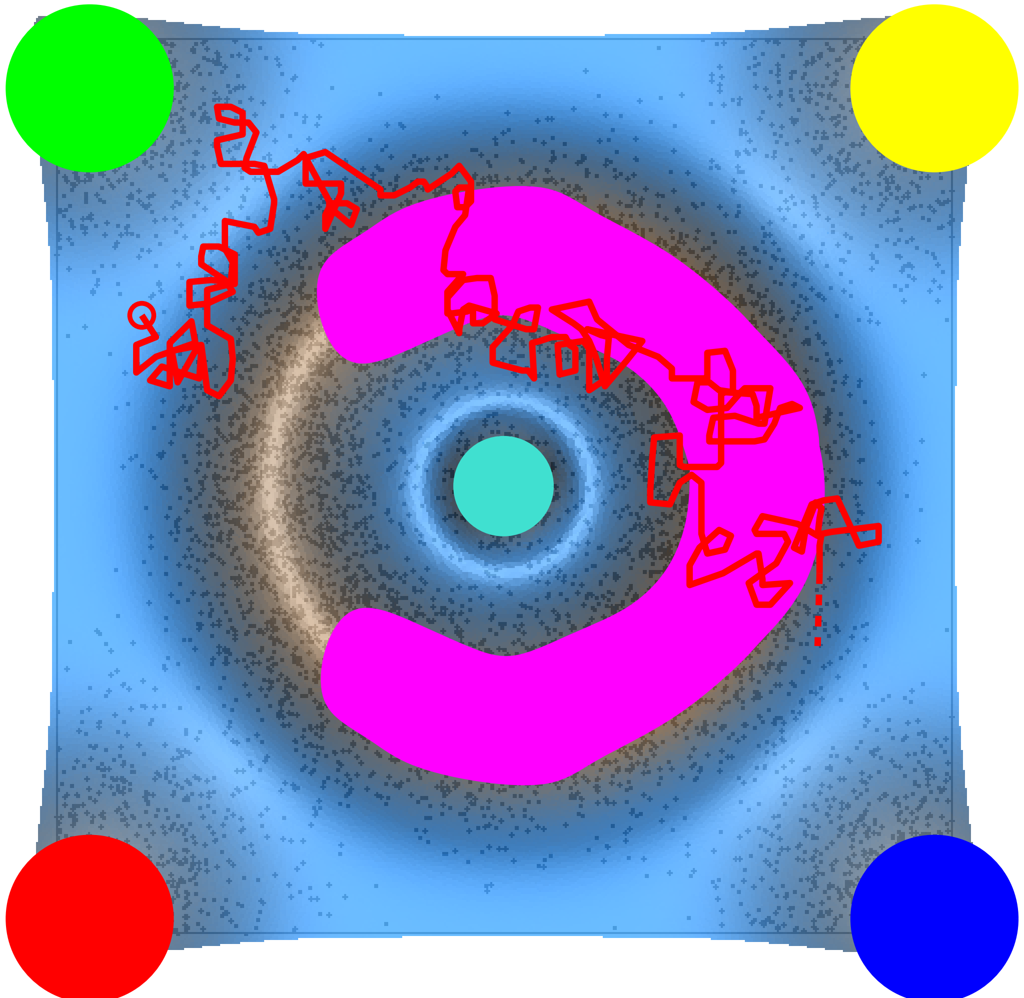
$\to$ solution: replace peaks by *cluster cores*:

- belong surely to the basin of attraction of a unique peak of f

- $\mathbb{P}(X_t$ eventually reaches some $C_i) > 0$

$\forall C_i$, $\mu_i(x) = \mathbb{P}(X_t$ reaches C_i first)

$\mu_0(x) = \mathbb{P}(X_t$ reaches none of the $C_i)$



Second idea (cont'd): the discrete setting

Input:

- $X_1, \cdots, X_n$ i.i.d. random variables drawn from the density f
- density estimator $\hat{f}_n$
- temperature parameter β
- estimators $\hat{C}_1, \cdots, \hat{C}_k$ of the (continuous) cluster cores $C_1, \cdots, C_k$

Second idea (cont'd): the discrete setting

Input:

- $X_1, \cdots, X_n$ i.i.d. random variables drawn from the density f
- density estimator $\hat{f}_n$
- temperature parameter β
- estimators $\hat{C}_1, \cdots, \hat{C}_k$ of the (continuous) cluster cores $C_1, \cdots, C_k$

Construction: Given $x \in X$, build the Markov chain $M^{h,n}$ s.t.:

- initial state = sample point nearest to x,
- transition kernel:

$$K^h(X_i, X_j) = \begin{cases} (1 + (\beta - 1)\frac{\hat{f}_n(X_i)}{\hat{f}_n(X_j)})Z_i & \text{if } \|X_i - X_j\|^2 \leq h \\ 0 & \text{otherwise} \end{cases}$$

where Z_i is the appropriate renormalization factor, so $\sum_{j=1}^n K^h(X_i, X_j) = 1$

Second idea (cont'd): guarantees

Hypotheses: (let $X = \{x \in \mathbb{R}^d \mid f(x) > 0\}$)

- f is C^1-continuous over $\mathbb{R}^d$
- $\lim_{\|x\|_2 \to +\infty} f(x) = 0$,
- $\forall \alpha_0 > 0$, $\exists \alpha < \alpha_0$ s.t. $\forall x \in X$, $f(x) = \alpha \Rightarrow \nabla f(x) \neq 0$,
- the SDE over X is well-posed,
- $\lim_{n \to \infty} \mathbb{P}(\|f - \hat{f}_n\|_\infty \geq \varepsilon) = 0$,
- $\forall \delta > 0$, $\lim_{n \to \infty} \mathbb{P}(C_i^{-\delta} \subseteq \hat{C}_i \subseteq C_i^{\delta}) = 1$.

Second idea (cont'd): guarantees

Hypotheses: (let $X = \{x \in \mathbb{R}^d \mid f(x) > 0\}$)

- f is C^1-continuous over $\mathbb{R}^d$ (regularity of the density)
- $\lim_{\|x\|_2 \to +\infty} f(x) = 0$,
- $\forall \alpha_0 > 0$, $\exists \alpha < \alpha_0$ s.t. $\forall x \in X$, $f(x) = \alpha \Rightarrow \nabla f(x) \neq 0$,

(to avoid leaving X)

- the SDE over X is well-posed,
- $\lim_{n\to\infty} \mathbb{P}(\|f - \hat{f}_n\|_\infty \geq \varepsilon) = 0$, (provided by estimator)
- $\forall \delta > 0$, $\lim_{n\to\infty} \mathbb{P}(C_i^{-\delta} \subseteq \hat{C}_i \subseteq C_i^{\delta}) = 1$. (provided by ToMATo)

Second idea (cont'd): guarantees

Hypotheses: (let $X = \{x \in \mathbb{R}^d \mid f(x) > 0\}$)

- f is C^1-continuous over $\mathbb{R}^d$ (regularity of the density)
- $\lim_{\|x\|_2 \to +\infty} f(x) = 0$,
- $\forall \alpha_0 > 0$, $\exists \alpha < \alpha_0$ s.t. $\forall x \in X$, $f(x) = \alpha \Rightarrow \nabla f(x) \neq 0$,

(to avoid leaving X)

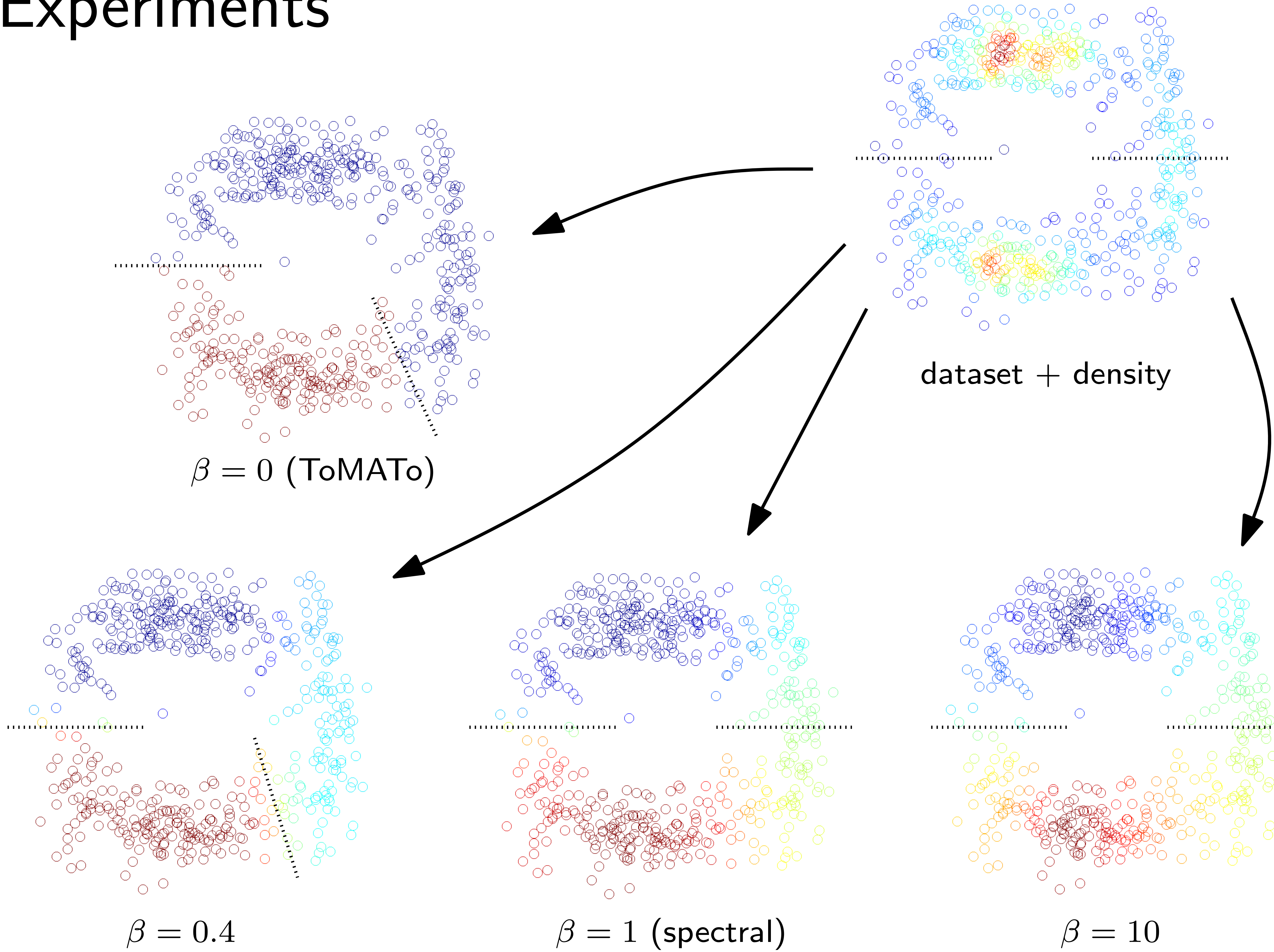
- the SDE over X is well-posed,
- $\lim_{n\to\infty} \mathbb{P}(\|f - \hat{f}_n\|_\infty \geq \varepsilon) = 0$, (provided by estimator)
- $\forall \delta > 0$, $\lim_{n\to\infty} \mathbb{P}(C_i^{-\delta} \subseteq \hat{C}_i \subseteq C_i^{\delta}) = 1$. (provided by ToMATo)

Conclusions:

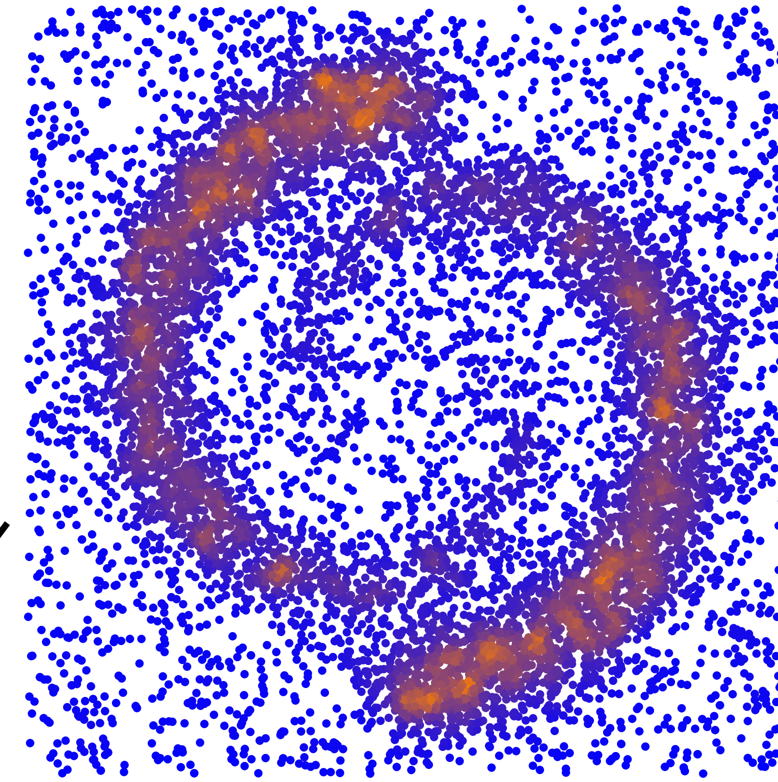
- the discrete process $M^{h,n}$ converges *weakly* to the solution of the SDE,

- the $\hat{\mu}_i^{h,n}$ derived from $M^{h,n}$ *converge in probability* to the μ_i derived from the solution of the SDE: $\forall U \subset X$ compact, $\forall \varepsilon > 0$, $\exists h_0$ s.t. $\forall h \leq h_0$, $\lim_{n\to\infty} \mathbb{P}\left(\sup_{x\in X} |\hat{\mu}_i^{h,n}(x) - \mu_i(x)| \geq \varepsilon\right) = 0$.

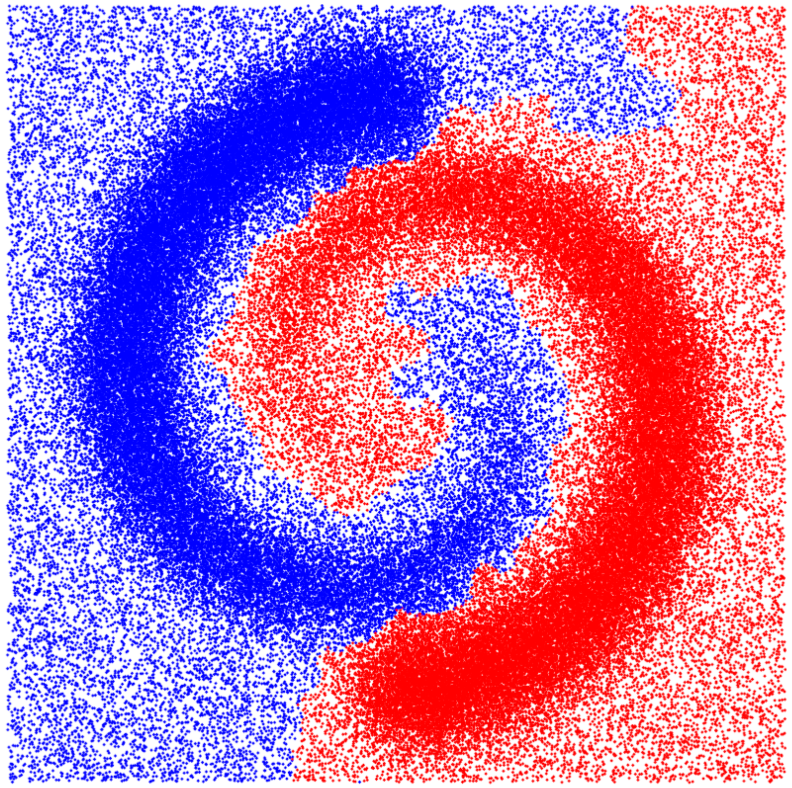
Experiments



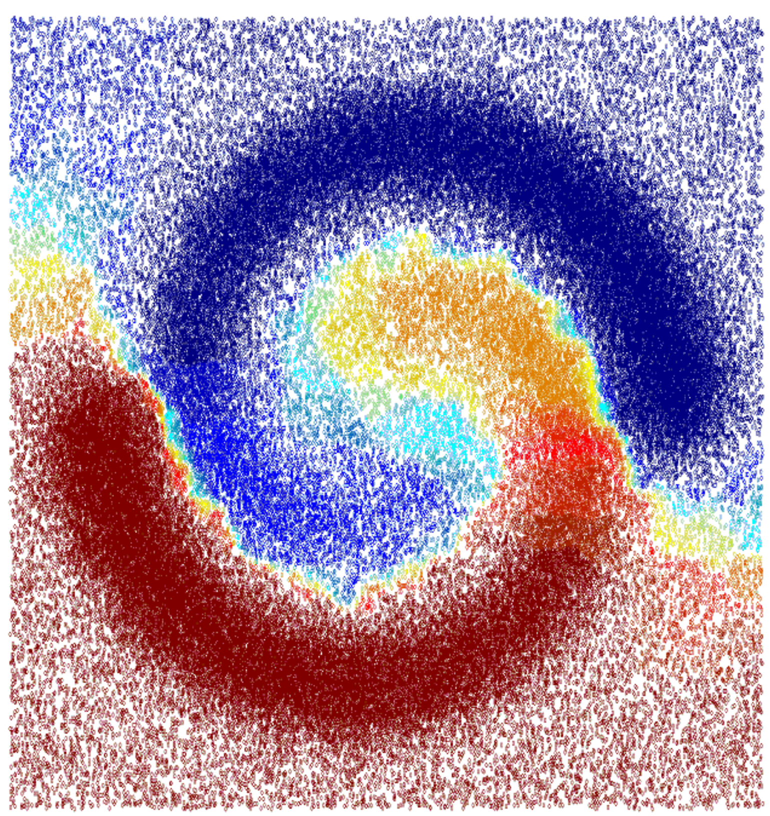
Experiments



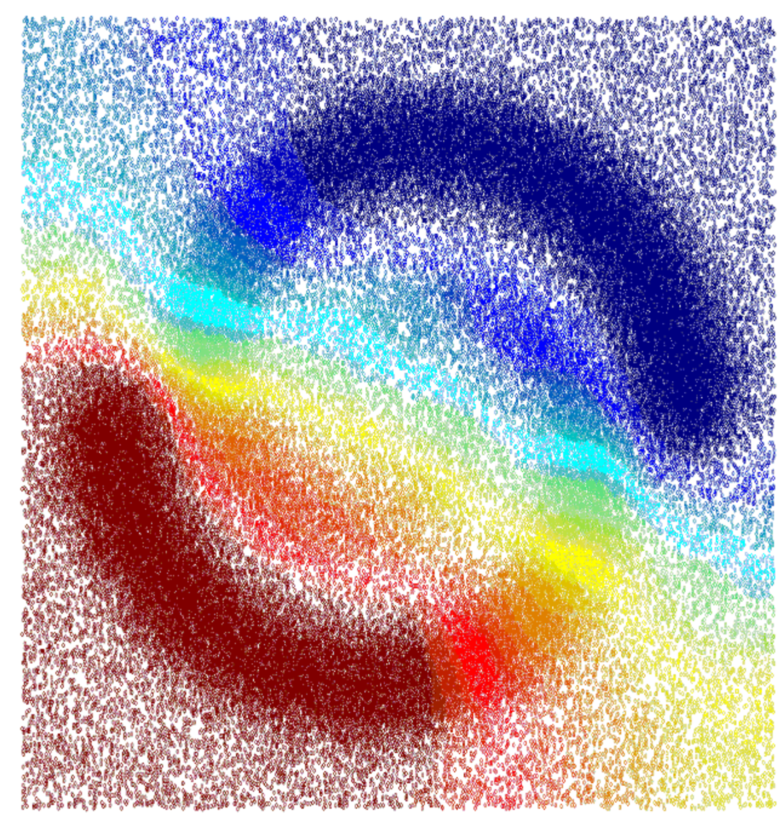
dataset $+$ density



$\beta = 0$ (ToMATo)

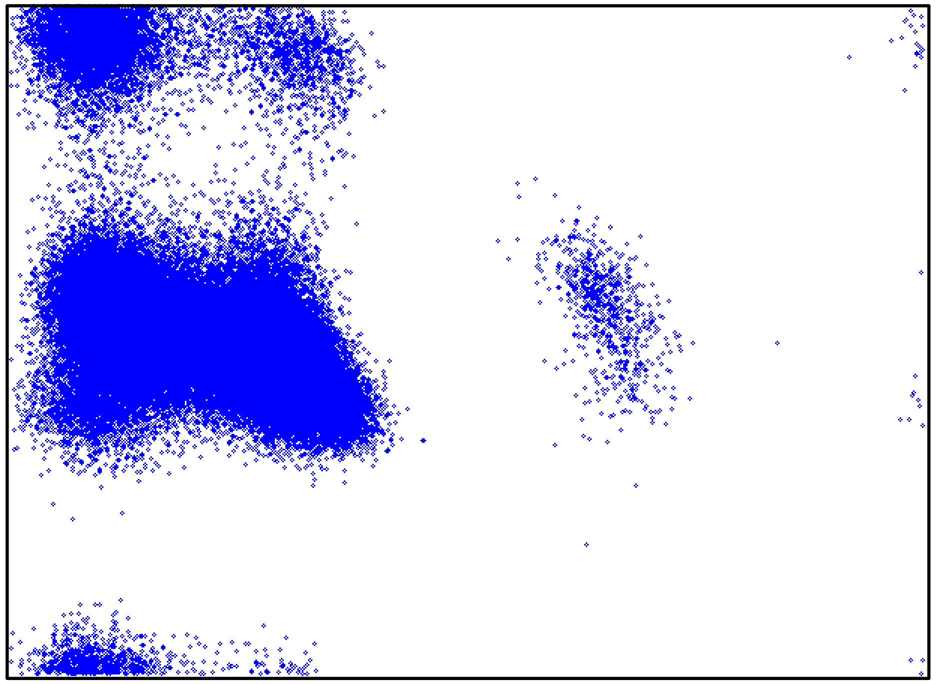


$\beta = 0.3$

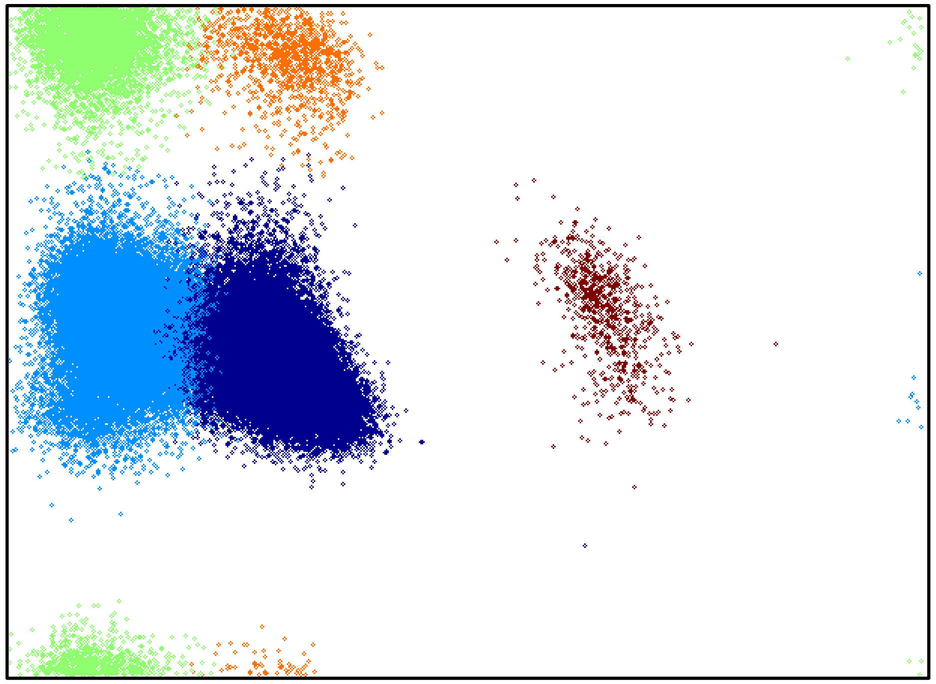


$\beta = 1$ (spectral)

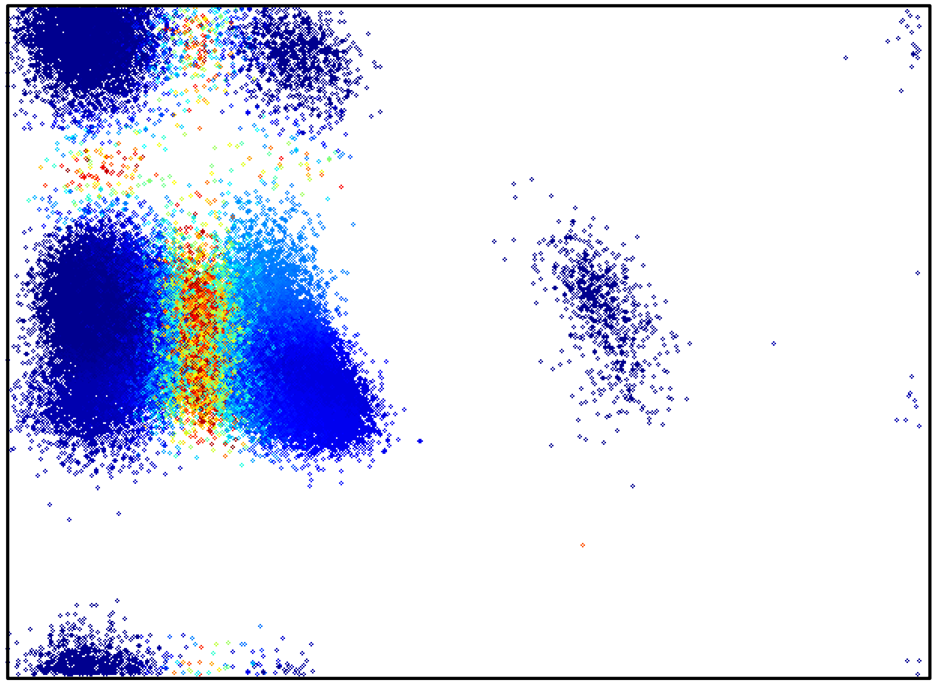
Experiments



21-d dataset (projected onto T^2)



hard clusters (ToMATo)



$\beta = 0.2$

Weak Convergence

- "the discrete process $M^{h,n}$ converges *weakly* to the solution of the SDE,"

Formally: given a fixed h, there is an $s(h)$ such that $M^{h,n}_{s\lfloor t/s \rfloor}$ converges weakly to the solution X_t of the SDE, i.e.:
For any $T, \varepsilon > 0$, for any $U \subset X$ compact, for any Borel set B in the Skorokhod space of trajectories $D([0,T], \mathbb{R}^d)$, there exists $h_0 > 0$ such that $\forall h \leq h_0$, $\lim_{n\to\infty} \mathbb{P}\left(\sup_{x\in U} |\mathbb{P}(M^{x,n,h}_{s\lfloor t/s \rfloor} \in B) - \mathbb{P}(X^x_t \in B)| \geq \varepsilon\right) = 0.$