TDA Update

What I’ve been doing, why I’ve been doing it,
and what the possible applications are

March 12, 2018
Matt Piekenbrock

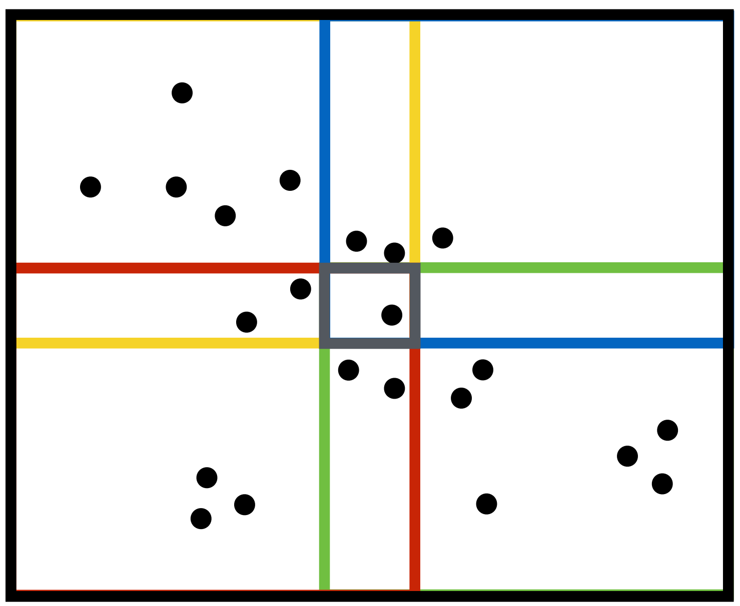
Mapper: Background

- We all know how mapper works…
- Algorithmically:
  1. Define a reference map $f : X \to Z$
  2. Construct a covering $\{U_\alpha\}_{\alpha \in A}$ of Z
     - A is called the index set
  3. Construct the subsets X_α
  4. Apply a clustering algorithm $\mathcal{C}$ to the sets X_α
  5. Obtain a cover $f^*(\mathcal{U})$ of X by considering the path-connected components of $f^{-1}(U_\alpha)$
     - Clusters form "nodes" / 0-simplexes
     - Non-empty intersections form "edges" / 1-simplexes
  6. The Mapper construction is the nerve of this cover, i.e.

$$M(\mathcal{U}, f) := N(f^*(\mathcal{U}))$$

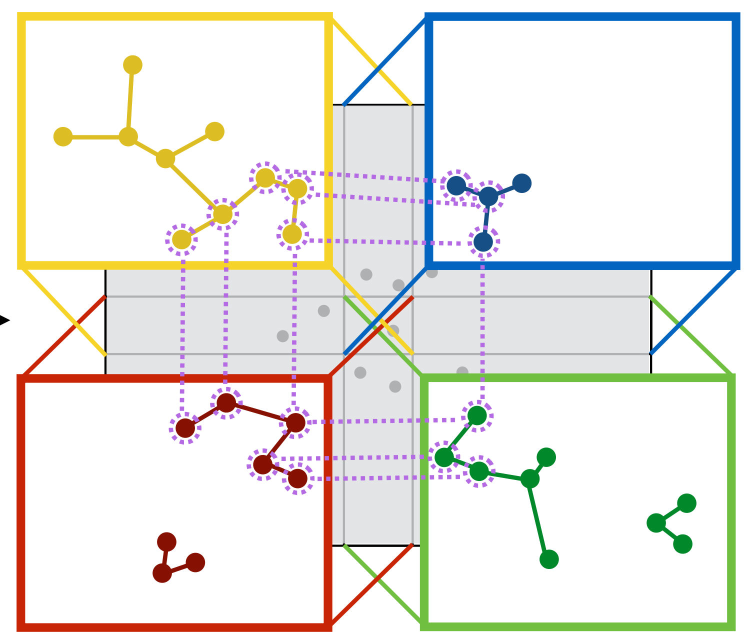
Mapper: Background

$f : X \to Z$



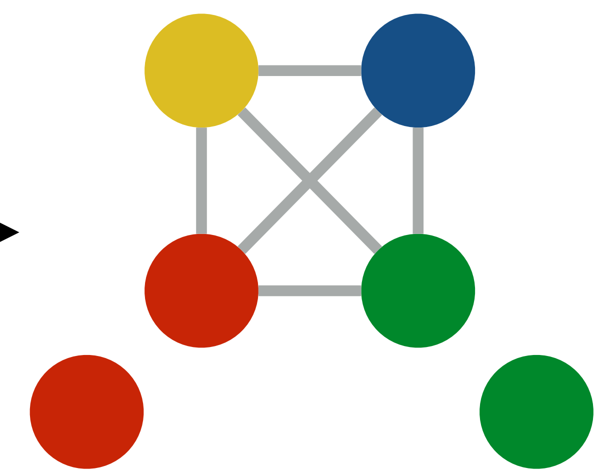
$\mathcal{U} = \{U_\alpha\}_{\alpha \in A}$

$X_\alpha = f^{-1} U_\alpha$



$C(X_\alpha) \to f^*(\mathcal{U})$

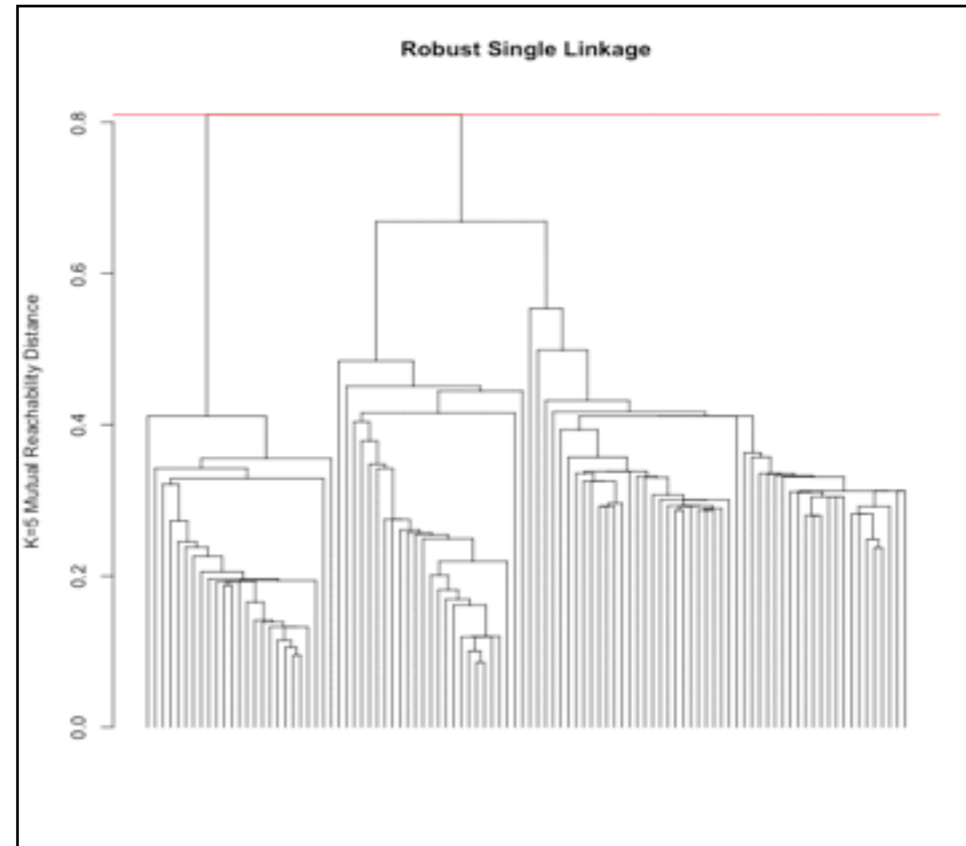
$f^*(\mathcal{U}) \to M(\mathcal{U}, f)$



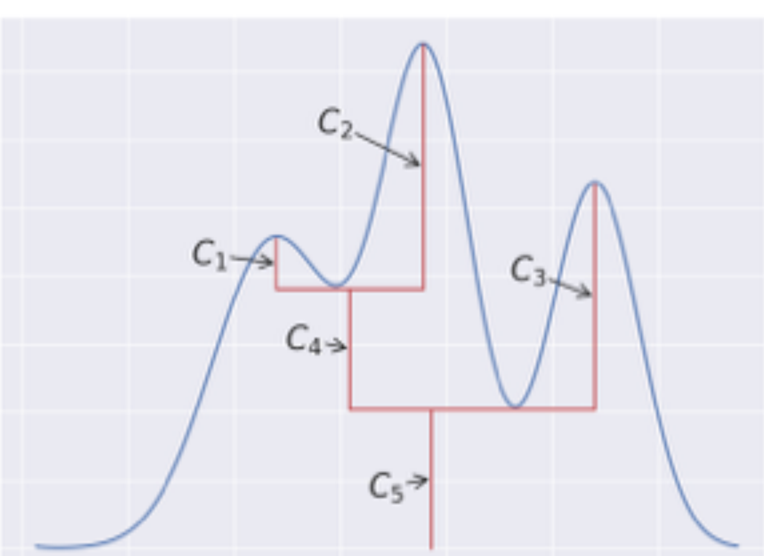
(1-skeleton example)

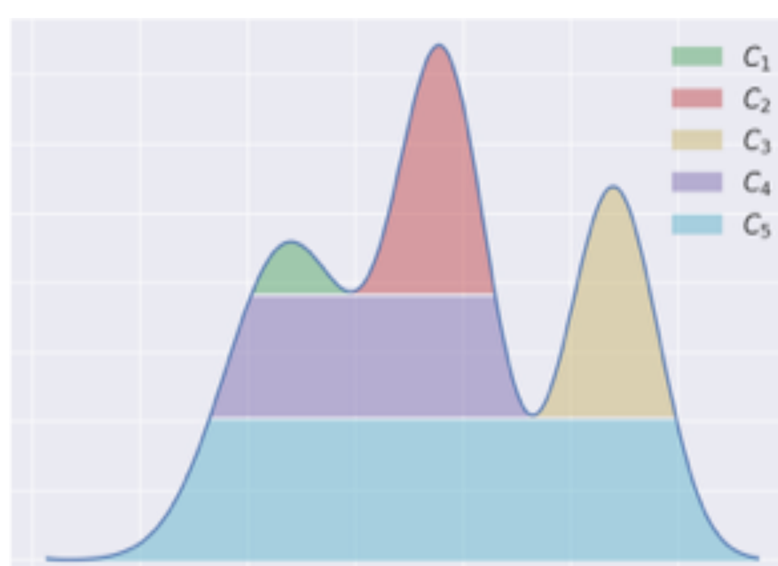
My Background

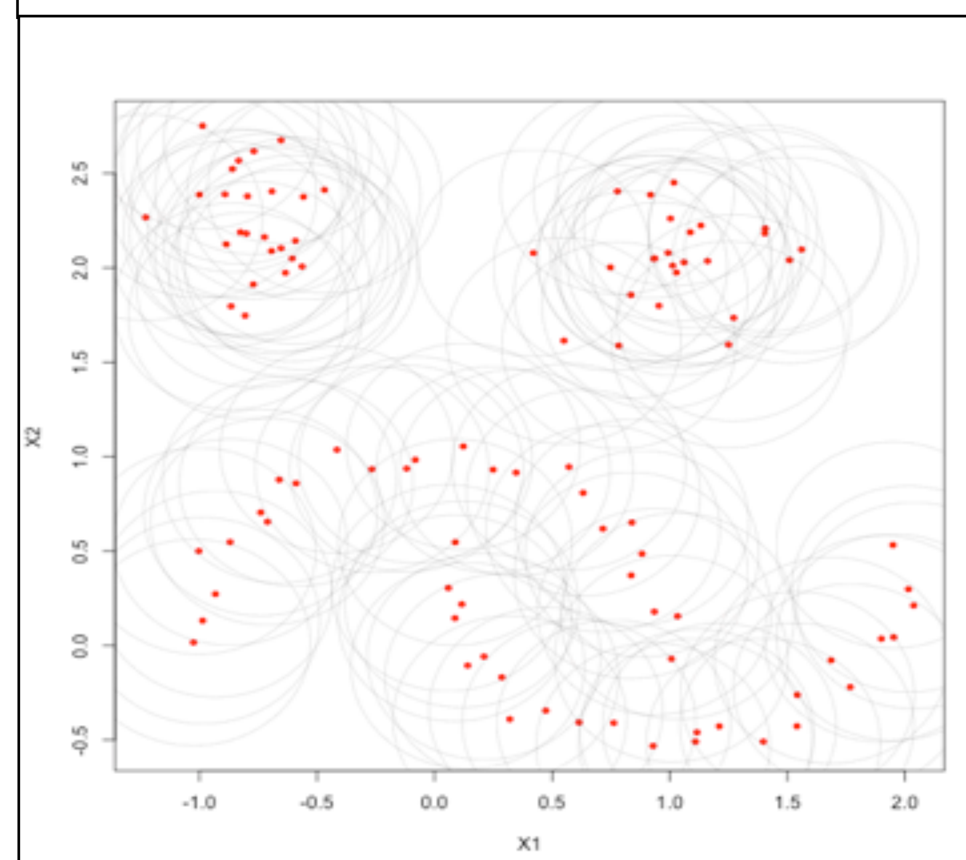
- Original goal: identify (or work towards…) creating some notion of **stability** within Mapper
- Background: Density-based clustering
  - To cluster things at multiple scales, **need to understand who structure evolves across parameter ranges**
  - Popular stability-based measure based on observing persistence of modes across density level threshold



$$S(\mathbf{C}_i) = \sum_{\mathbf{x}_j \in \mathbf{C}_i} \left(\lambda_{\max}(\mathbf{x}_j, \mathbf{C}_i) - \lambda_{\min}(\mathbf{C}_i) \right) = \sum_{\mathbf{x}_j \in \mathbf{C}_i} \left(\frac{1}{\varepsilon_{\min}(\mathbf{x}_j, \mathbf{C}_i)} - \frac{1}{\varepsilon_{\max}(\mathbf{C}_i)} \right)$$





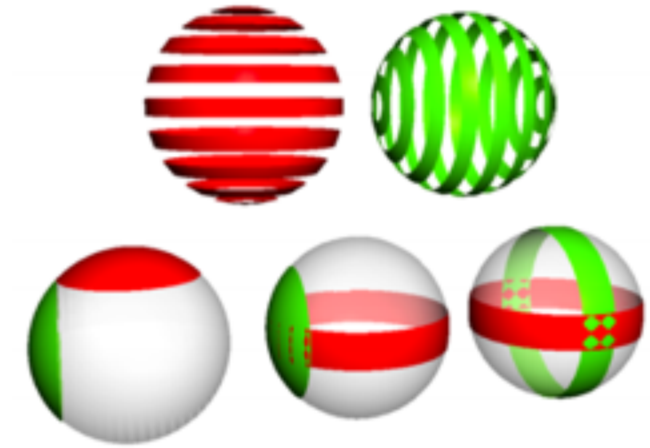


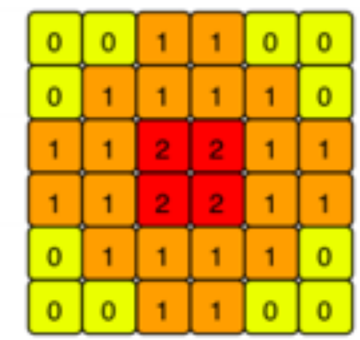
Mapper: Background

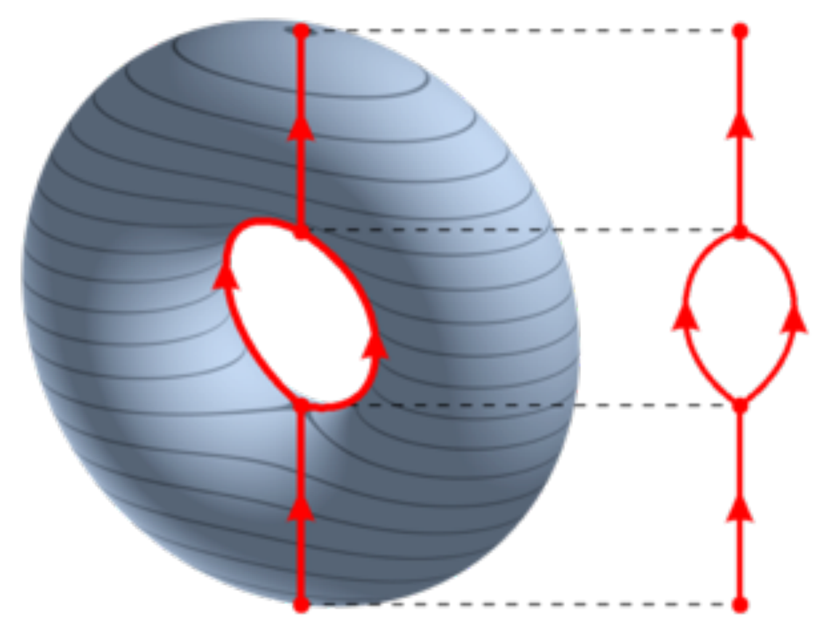
- What does mapper actually do?
  - “[Mapper] takes as input both a possibly high dimensional dataset and a map defined on the data, and produces a summary of the data by using a cover of the codomain of the map. This cover, via a pullback operation to the domain, produces a simplicial complex connecting the data points.” - Dey et. al [Multiscale Mapper]
  - “In the case where the target parameter space is R, our construction amounts to a stochastic version of the Reeb graph associated with the filter function. If the covering of R is too coarse, we will be constructing an image of the Reeb graph of the function, while if it is fine enough we will recover the Reeb graph precisely.” - Singh. & Carlsson et. al [Mapper]

Background: Reeb Graphs

- A Reeb graph is a mathematical object reflecting the evolution of the level sets of a real-valued function on a manifold. - Wikipedia
- Reeb space == multivariate generalization of Reeb graph
  - "…**compresses the components of the level sets** of a multivariate and obtains a summary representation of their relationships"
- Munch et. al proved that the categorical representations of the Reeb space and Mapper converge in terms of interleaving distance







$$L_c(f) = \{(x_1, \cdots, x_n) \mid f(x_1, \cdots, x_n) = c\} \ ,$$

“From Clouds to Complexes”

- Point cloud —> simplicial complex
  - Combinatorial graph where nodes represent summaries of data
  - edges represent proximity
- Well-known methods for computing a simplicial complex
  - Vietoris-Rips complex
  - Cech Complex

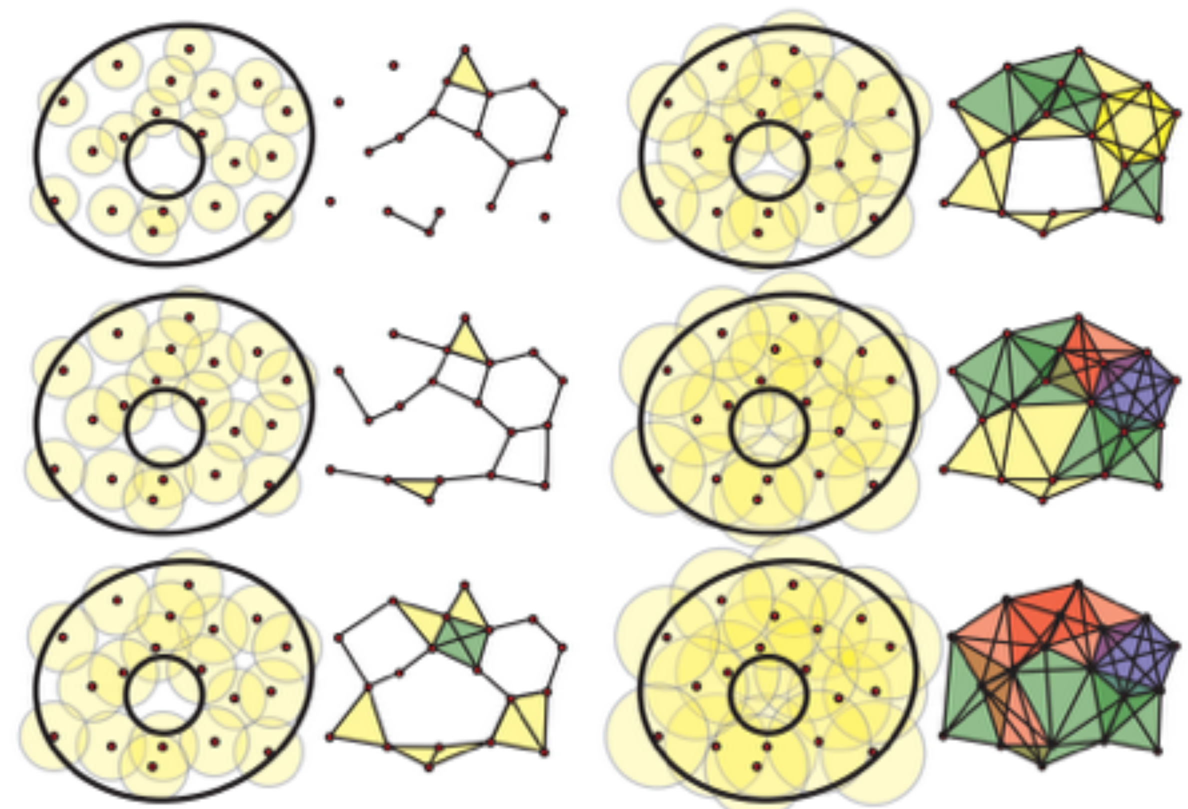


FIGURE 3. A sequence of Rips complexes for a point cloud data set representing an annulus. Upon increasing ϵ, holes appear and disappear. Which holes are real and which are noise?

Persistent Homology

- Recurring theme in applied topological data analysis
- "Despite being both computable and insightful, the homology of a complex associated to a point cloud at a particular ϵ is **insufficient**: it is a mistake to ask which value of ϵ is optimal." - Ghrist

- "The motivation is that, for a parameterized family of spaces (i.e. VR complexes) modeling a point-cloud data set, **qualitative features which persist over a large parameter range have greater statistical significance**"

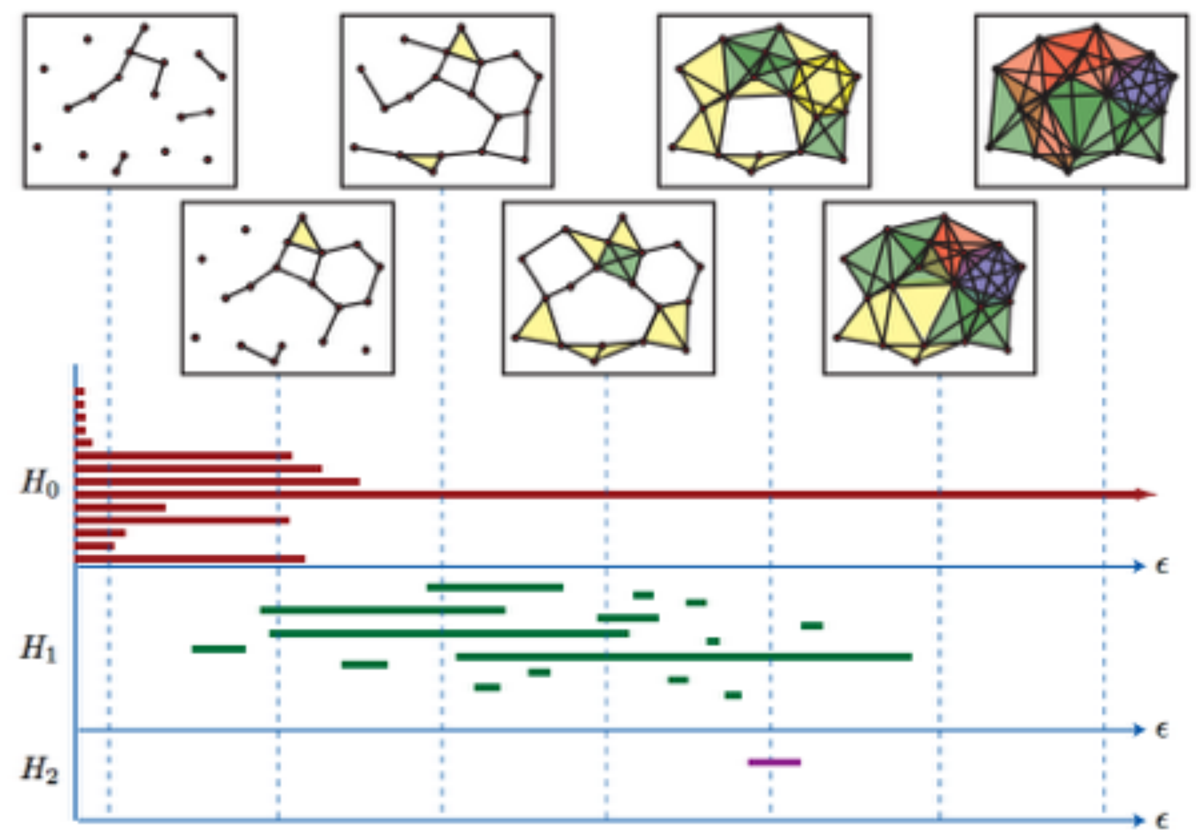


FIGURE 4. [bottom] An example of the barcodes for $H_*(\mathsf{R})$ in the example of Figure 3. [top] The rank of $H_k(\mathcal{R}_{\epsilon_i})$ equals the number of intervals in the barcode for $H_k(\mathsf{R})$ intersecting the (dashed) line $\epsilon = \epsilon_i$.

Studying Mapper: In the context of Persistent Homology

- Is it possible to study Mapper in the context of Persistent Homology? (or something close to it)
- "The icon of persistence is a monotone sequence

$$\cdots \rightarrow \cdot \rightarrow \cdot \rightarrow \cdots$$

where arrows connote maps of spaces or chains or the induced homomorphisms on homology." - Ghrist
- So, we just need some kind of monotone sequence between Mapper complexes?

Let K and L be two finite simplicial complexes over the vertex sets V_K and V_L, respectively. A set map $\phi : V_K \to V_L$ is a *simplicial map* if $\phi(\sigma) \in L$ for all $\sigma \in K$.

Multi-scale Mapper

- “The resulting view of the data [produced by Mapper] through a cover of the codomain offers flexibility in analyzing the data. However, *it offers only a view at a fixed scale at which the cover is constructed*.”
- Dey et. al introduce *Multiscale mapper,* a “tower” of simplicial complexes, which is a chain of simplicial complexes connected by simplicial mapping
  - Nice benefit: if the map is a real-valued PL function, the *exact* persistence diagram *from only the 1-skeleton* (!)
- “Interestingly, analogous to the case of homology versus persistence homology, mapper does not satisfy a stability property, whereas multiscale mapper *does* enjoy stability as we show in this paper.”

(Possible stop here)

Mapper: Complexity

1. Define a reference map $f : X \to Z$
2. Construct a covering $\{U_\alpha\}_{\alpha \in A}$
3. Apply a clustering algorithm $\mathcal{C}$ to the sets X_α
4. Obtain a cover $f^*(\mathcal{U})$ of X by considering the path-connected components of $f^{-1}(U_\alpha)$
5. The Mapper construction is the nerve of this cover, either the:
   1. 1-skeleton
   2. *n*-skeleton

$$(\text{e.g. } O(n^2))$$
$$+$$
$$O(nd)$$
$$+$$
$$O(n^2) + O(n^2) + O(n\alpha)$$
$$+$$
$$O(n\alpha)$$
$$+$$
$$O(n^3)$$
or
$$O(3^{n/3} \times n^2)$$

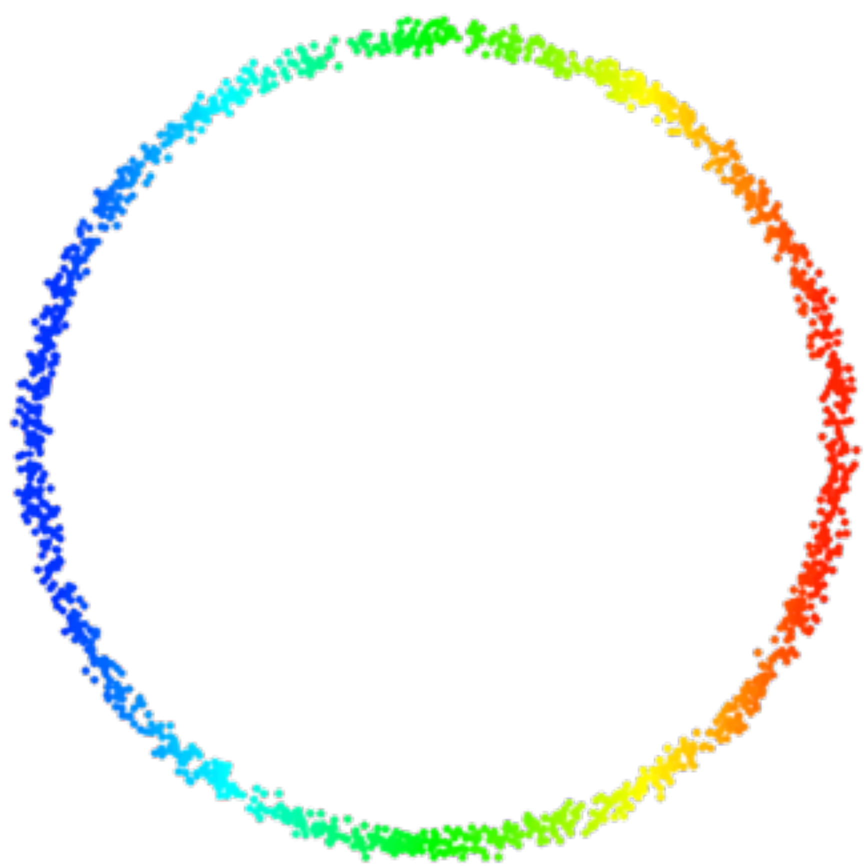
Parameters

- Mapper has several parameters
  - Filter function is <u>very important</u>, but generally application specific
  - Clustering algorithm / hyper-parameters is (perhaps) of minor importance
- Choice of cover is significant
  - Resolution (number of intervals) ~= nodes
  - Gain (percent overlap) ~= connectivity (!!!)
- Percent overlap **determines entire connectivity** of the graph

Problem: How to compute?

- While such theoretical results are great, how would one actually compute Multiscale Mapper (MM)?
- Consider the following:

$$\boxed{X = \{x_1, x_2, \ldots, x_n\}}$$

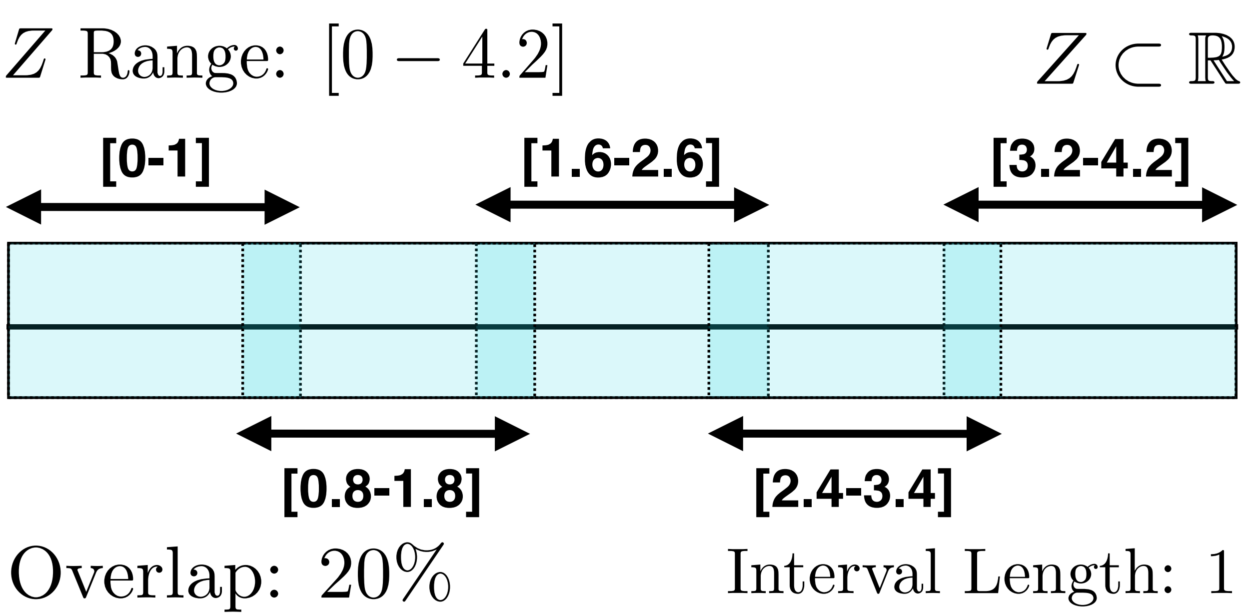


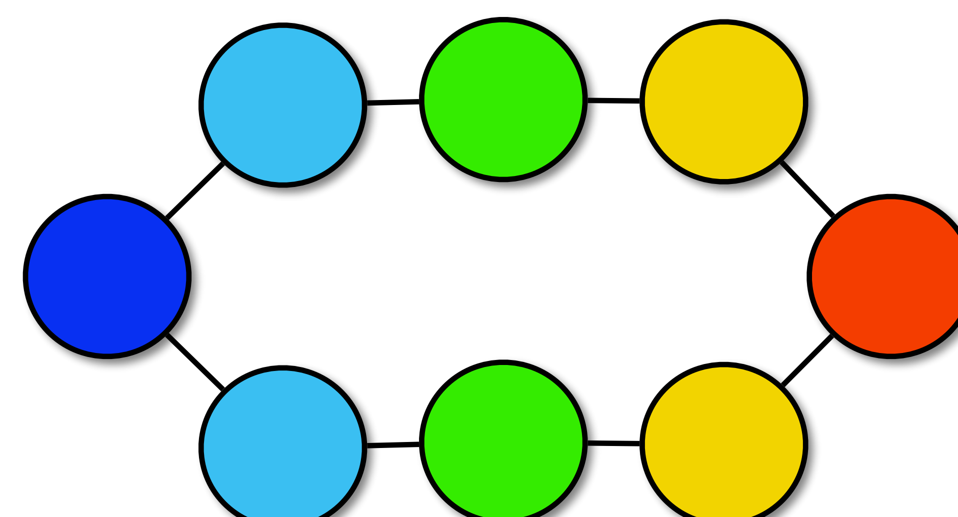
Problem: How to compute?

- While such theoretical results are great, how would one actually compute Multiscale Mapper (MM)?
- Consider the following:

$$\boxed{f(x) = \|x - p\|_2}$$

$$\boxed{G(V, E)}$$

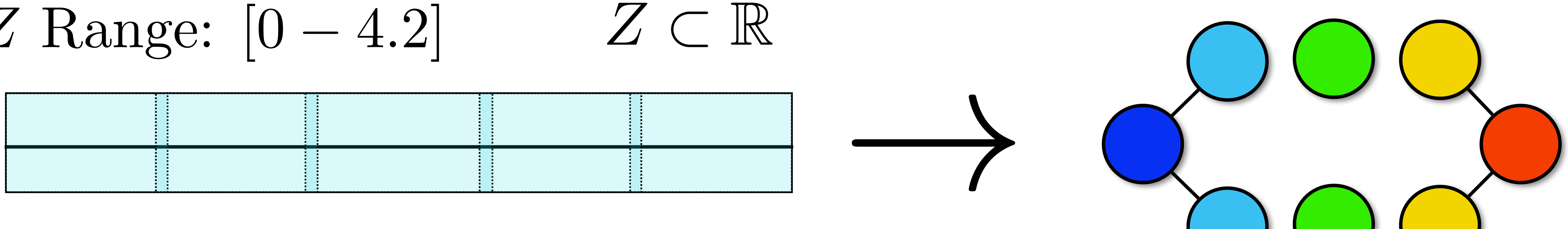




Problem: How to compute?

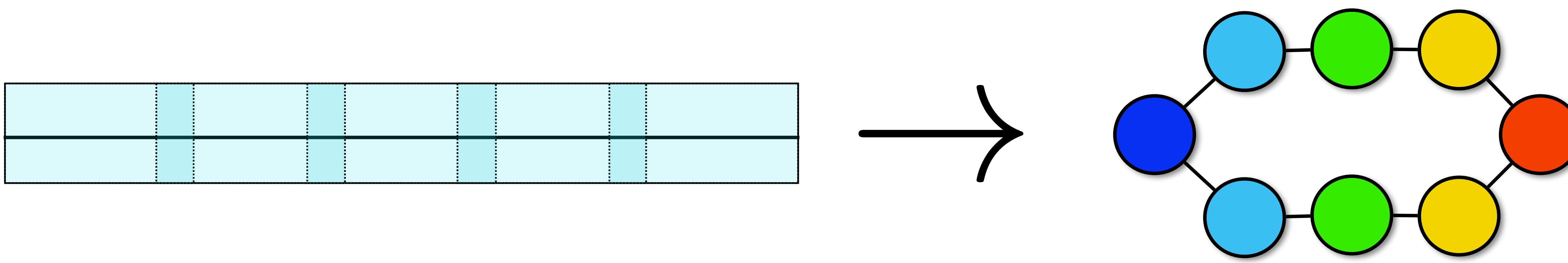
$\boxed{f(x) = \|x - p\|_2}$ $\boxed{G(V, E)}$

Z Range: $[0 - 4.2]$ $Z \subset \mathbb{R}$



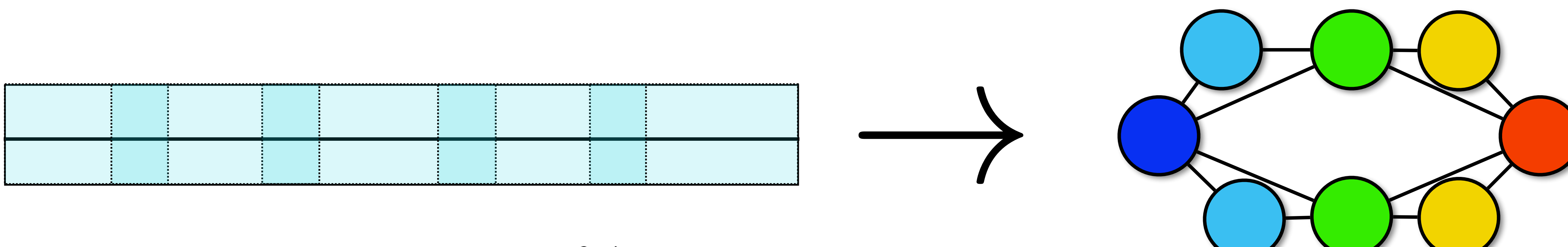
Overlap : 5%

$\epsilon \to \infty$



Overlap: 20%

$\epsilon \to \infty$



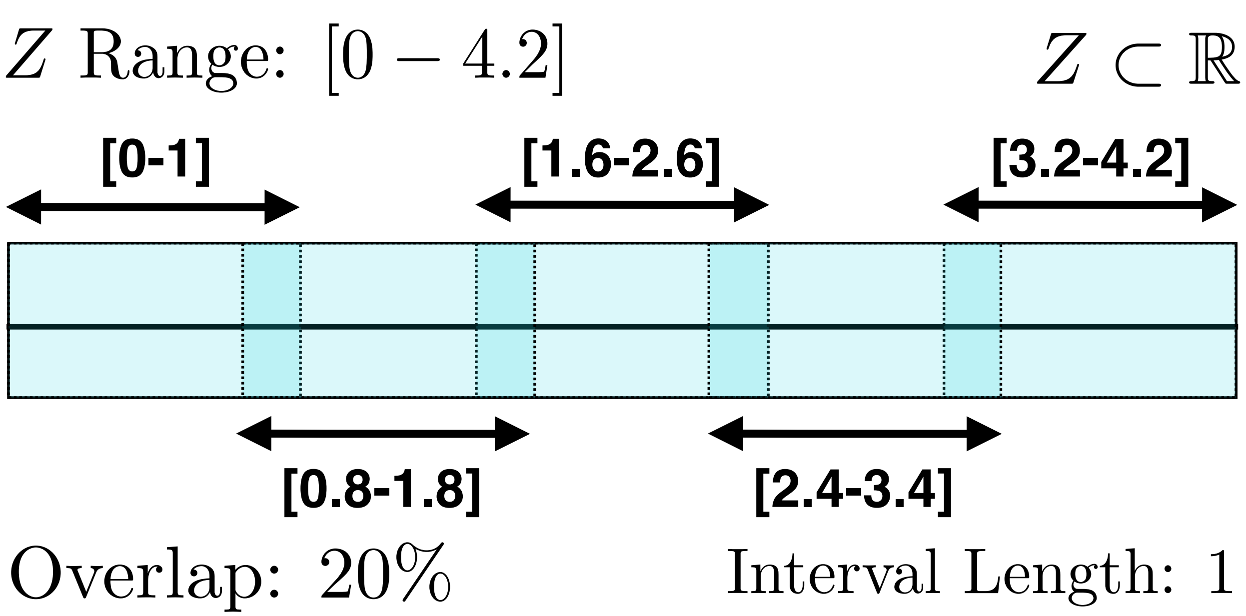
Overlap : 40%

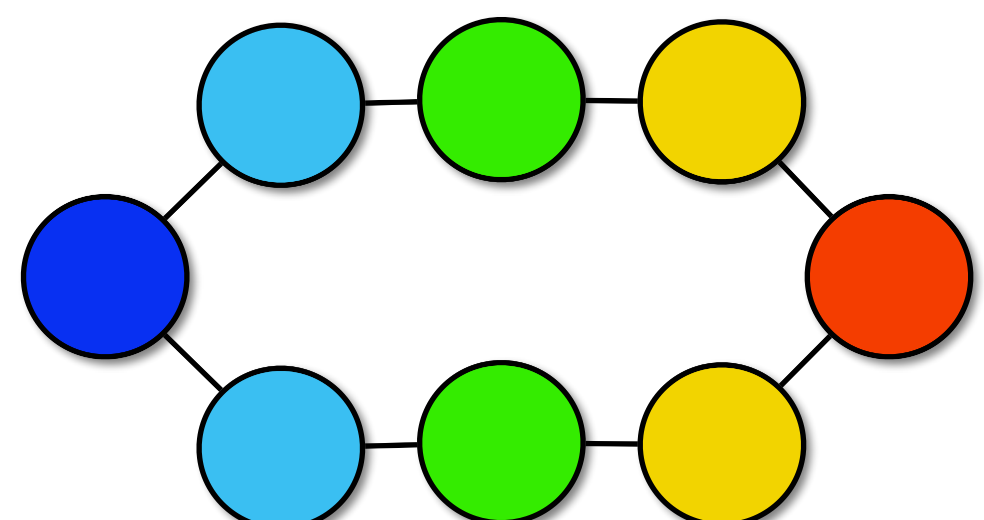
Problem: How to compute?

- While such theoretical results are great, how would one actually compute Multiscale Mapper (MM)?
- Consider the following:

$$f(x) = \|x - p\|_2$$

$$G(V, E)$$

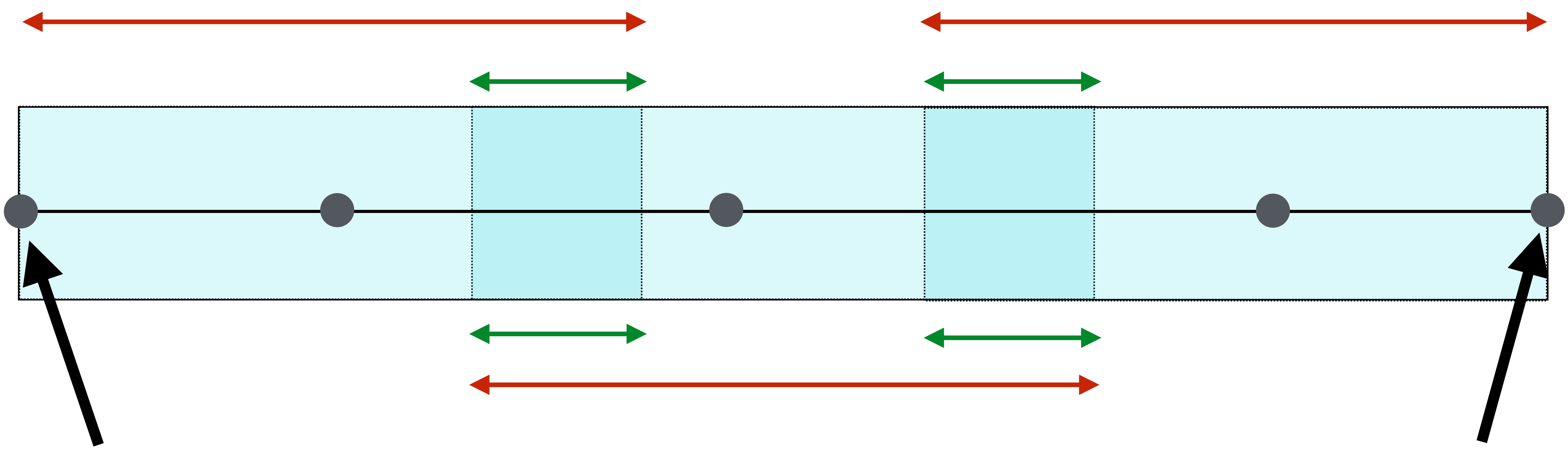




Consider the problem

$$f : X \to Z$$

$$\bar{\mathbf{z}} = \mathbf{z}_{max} - \mathbf{z}_{min}$$

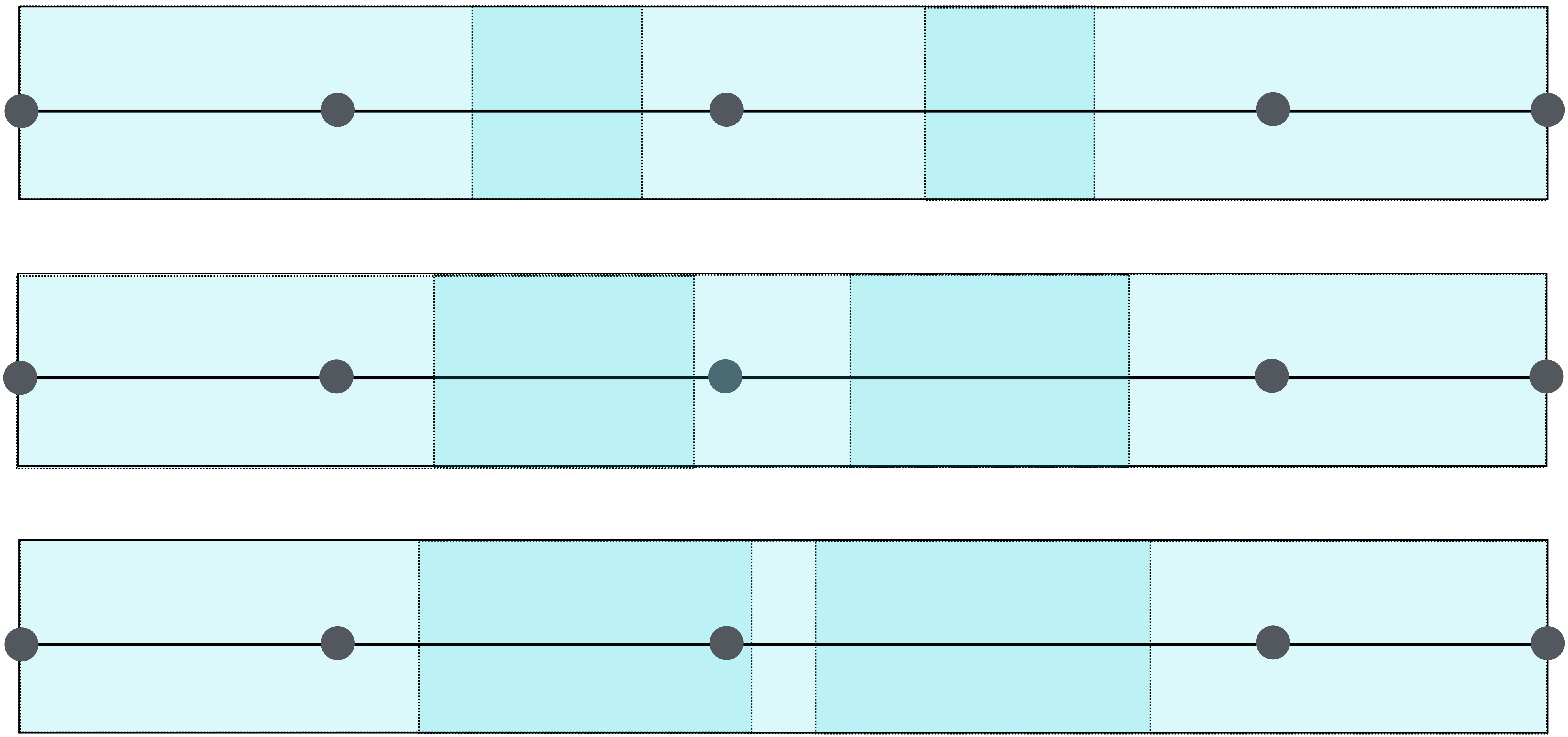


$\mathbf{z}_{min}$ $\qquad$ $\mathbf{k} = 3, \mathbf{g} = 0.20$ $\qquad$ $\mathbf{z}_{max}$

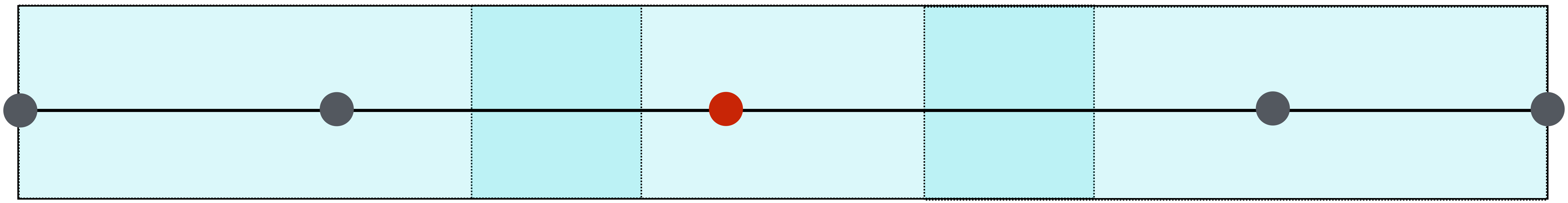
$$\mathbf{r} = \frac{\bar{\mathbf{z}}}{(\mathbf{k} - \mathbf{g}(\mathbf{k} - 1))} \qquad \mathbf{e} = \mathbf{r} \circ (1 - \mathbf{g})$$

Consider the problem

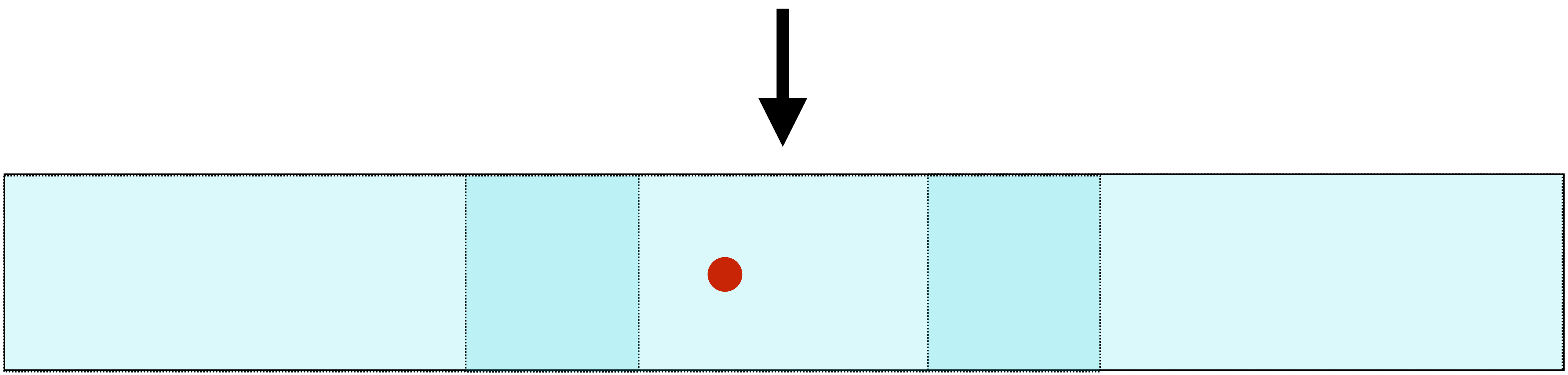
What is the smallest overlap value that could induce a new simplicial complex?



Consider the problem

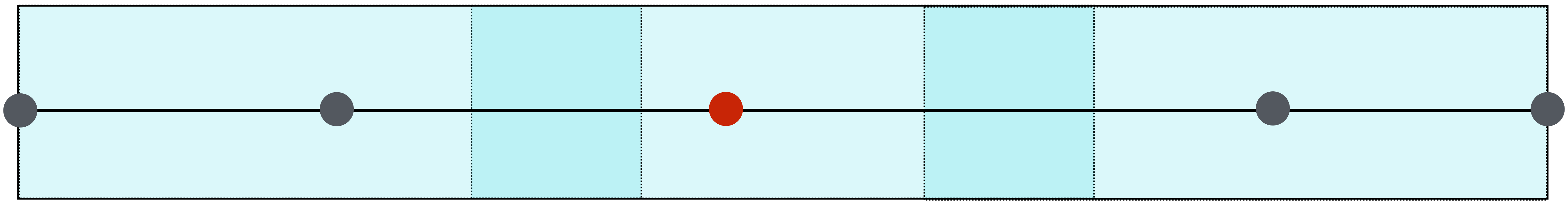


What is the smallest overlap values that *this point* could potentially induce a new complex?

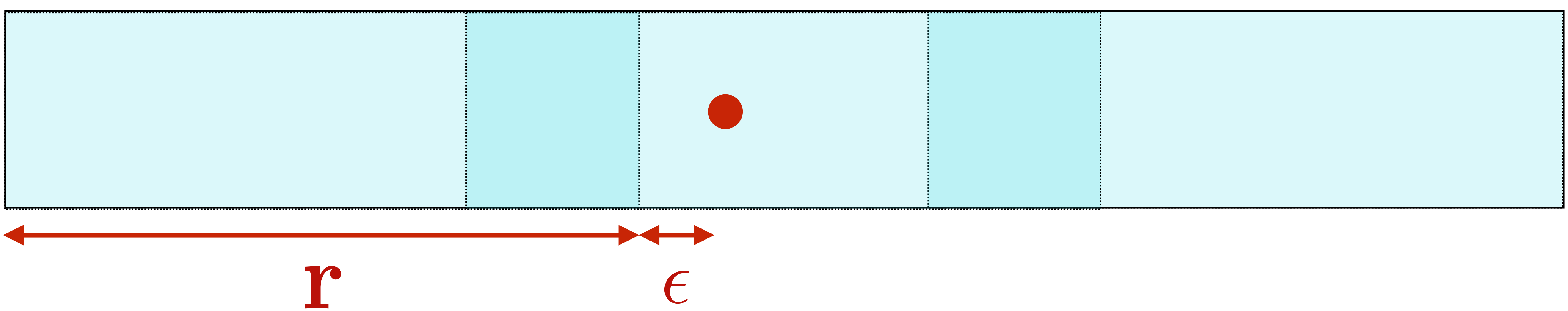


What interval length will this point intersect a new level set?

Consider the problem



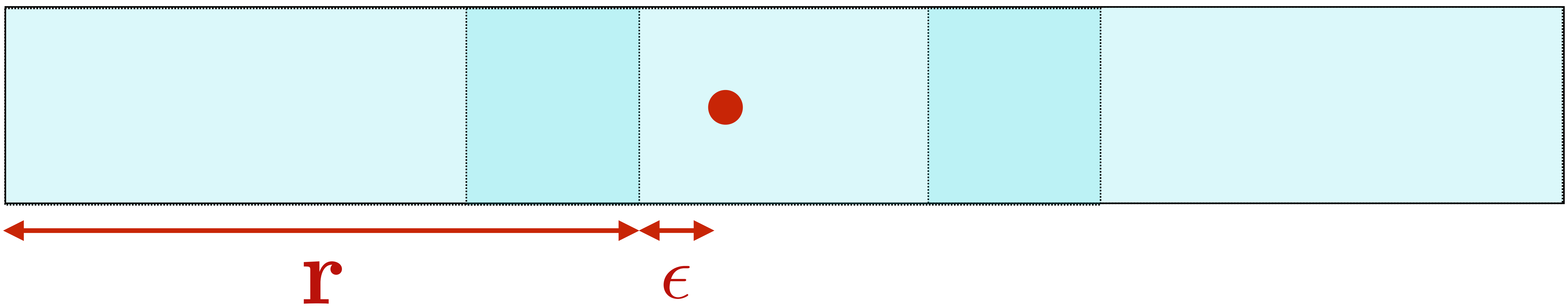
What is the smallest overlap values that *this point* could potentially induce a new complex?



Equivalent: What interval length will this point intersect a new level set?

Consider the problem

Equivalent: What interval length will this point intersect a new level set?



Recall: $\mathbf{r} = \dfrac{\bar{\mathbf{z}}}{(\mathbf{k} - \mathbf{g}(\mathbf{k}-1))}$ $\qquad \mathbf{e} = \mathbf{r} \circ (1 - \mathbf{g})$

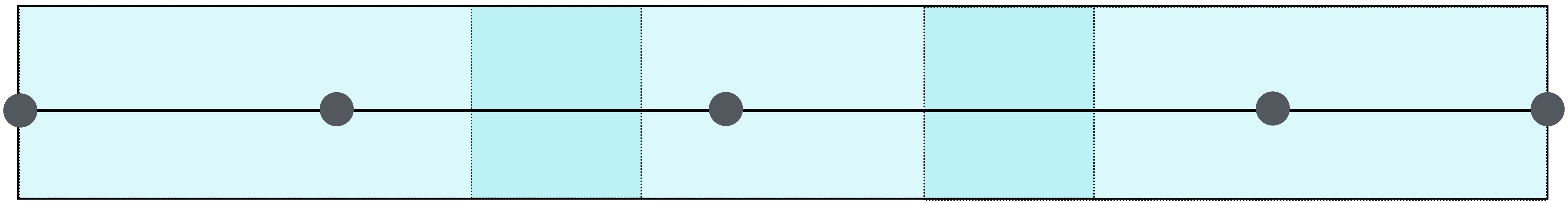
↓

Idea: $\mathbf{g} = \dfrac{\bar{\mathbf{z}} - \mathbf{r}\mathbf{k}}{\mathbf{r}(-\mathbf{k}+1)}$

Observation: If *k* is fixed, just need *r*!

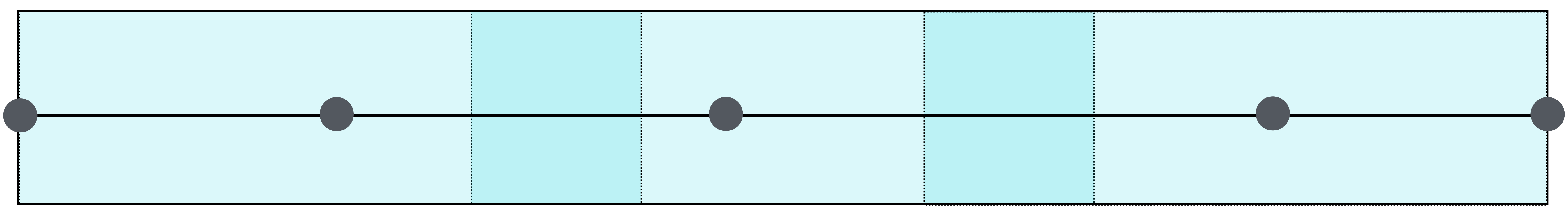
Consider the problem

Problem: Keeping track of each level set may be computationally difficult



Observation: Each point is already associated with an index…

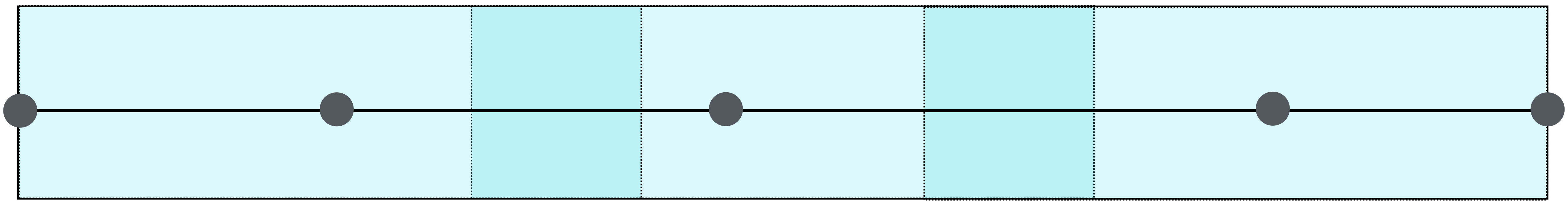
$$\{U_\alpha\}_{\alpha \in A}$$



$\alpha_0 = (0)$ $\alpha_1 = (1)$ $\alpha_2 = (2)$

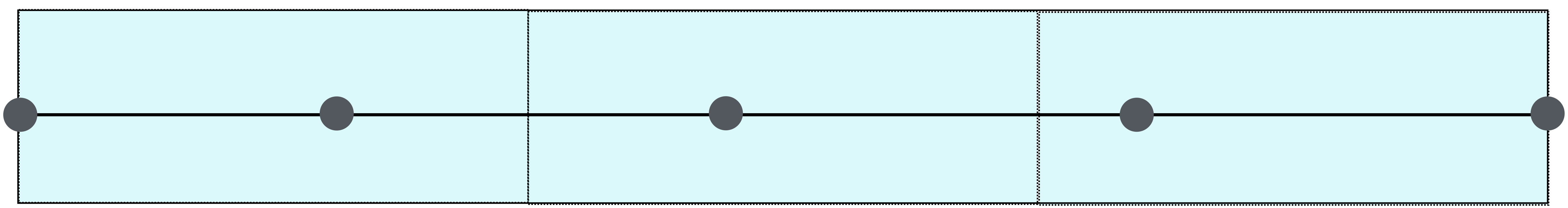
Idea: Each point is already associated with an index…

Observation: Each point is already associated with a set of indexes… $\{U_\alpha\}_{\alpha \in A}$



$\alpha_0 = (0)$ $\alpha_1 = (1)$ $\alpha_2 = (2)$

When there is 0 overlap, there is only one index…

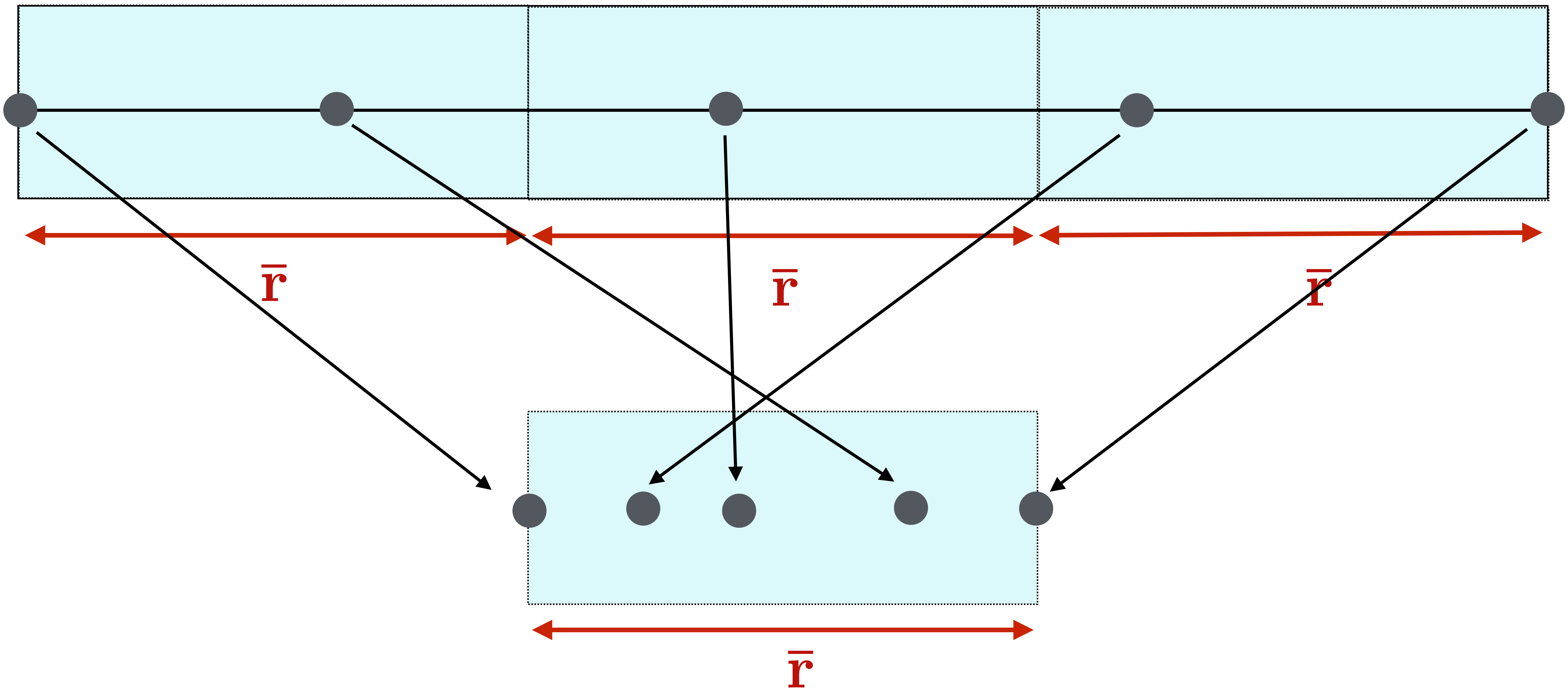


What if we record that index for each point?

$$\mathbf{A} = [\alpha_0, \alpha_0, \alpha_1, \alpha_2, \alpha_2]^\intercal \longrightarrow \mathbf{A} = [0, 0, 1, 2, 2]^\intercal$$

And start with this *base* interval length

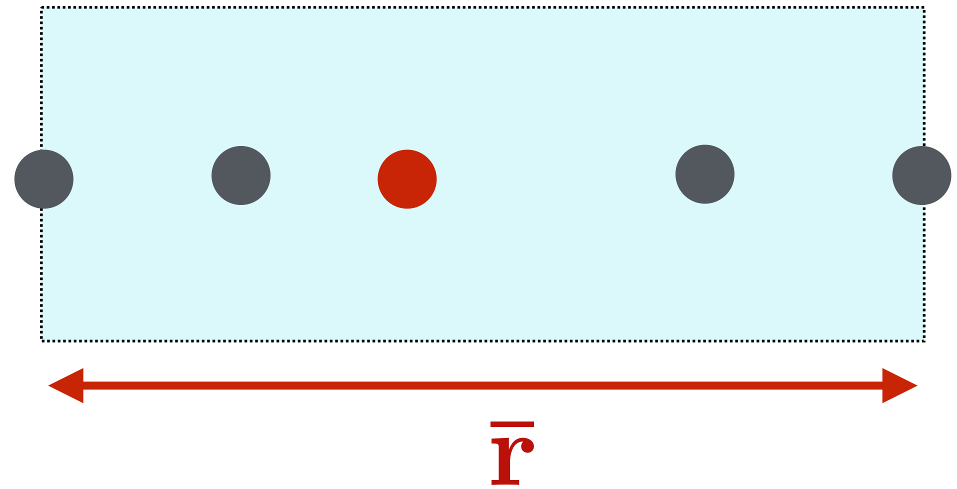
$$\mathbf{k} = 3, \mathbf{g} = 0 \longrightarrow \bar{\mathbf{r}} = \frac{\bar{\mathbf{z}}}{\mathbf{k}} \quad \mathbf{e} = 0$$



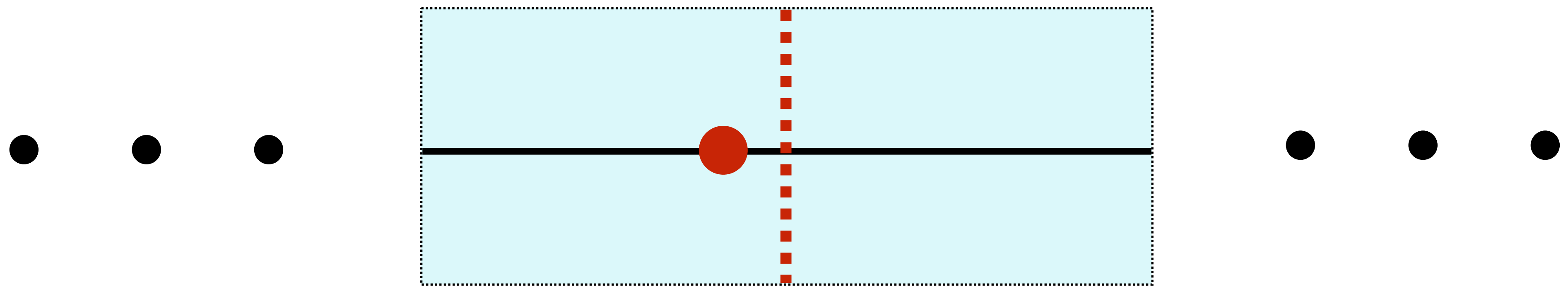
And project to a "unit" box…

$$\widetilde{\mathbf{Z}} = (\mathbf{Z} - \mathbf{z}_{\mathbf{min}}) - \mathbf{A} \circ \bar{\mathbf{r}}$$

And project to a "unit" box…

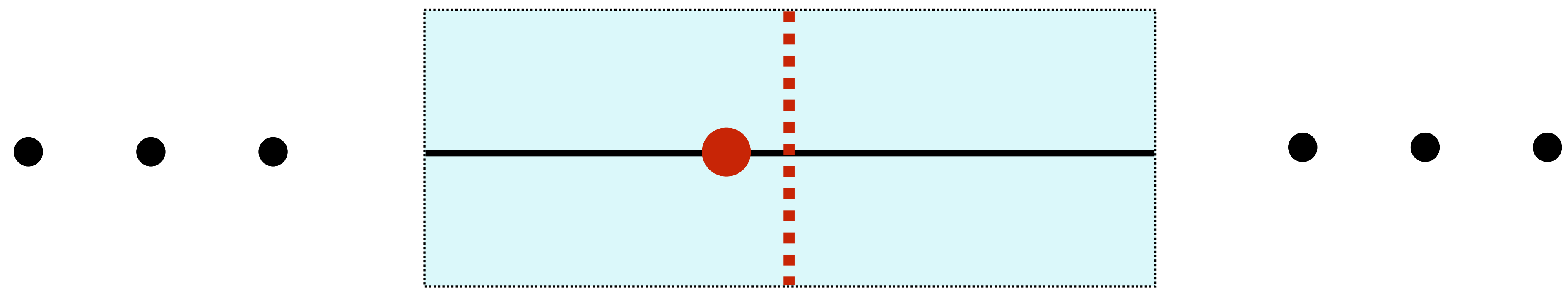


Goal: What's the distance to the closest box?



Observation: Boxes expand 'uniformly' in both directions
Just need the distance to the outside of the interval

$$\mathbf{z}_{\Delta} = [\ldots, \min(\widetilde{\mathbf{z}_\mathbf{i}}, \bar{\mathbf{r}} - \widetilde{\mathbf{z}_\mathbf{i}}), \ldots]^{\mathsf{T}}$$



The minimum interval length for each point to intersect its closest level set is thus:

$$\hat{\mathbf{R}} = \mathbf{Z}_{\Delta} + \bar{\mathbf{r}}$$

The corresponding overlap values?

$$\mathbf{G} = \frac{(\bar{\mathbf{z}} - \hat{\mathbf{R}}\mathbf{k})}{\hat{\mathbf{R}}(-\mathbf{k} + 1)}$$

Why is having all overlap values useful?

Consider Mappers complexity…

Filter function		Form cover		Distance Matrix	Cluster Hierarchically	Cut Tree
$f : X \to Z$	$+$	$\{U_\alpha\}_{\alpha \in A}$	$+$	$D(X_\alpha)$	$\mathcal{C}_H(X_\alpha)$	$\mathcal{C}(X_\alpha)$

$(\text{e.g. } O(n^2))$ $O(nd)$ $O(n^2) + O(n^2) + O(n\alpha)$

(needed every instance of Mapper) **(needed every instance of Mapper)** **(needed every instance of Mapper)**

	Form connected components		Form 1-skeleton		Form *n*-skeleton
$+$	$f^*(\mathcal{U})$	$+$	$M(\mathcal{U}, f)^{(1)}$	or	$M(\mathcal{U}, f)^{(k)}$

$O(n\alpha)$ $O(n^3)$ $O(3^{n/3} \times n^2)$

(needed every instance of Mapper) **(needed every instance of Mapper)** **(needed every instance of Mapper)**

Why is having all overlap values useful?

- Consider the following strategy instead
  - Construct “base” cover
  - Compute all overlap values, sort by increasing value
    - We know which level sets each point will intersect (and “when”)
  - Update only level sets that need updating
  - Update only simplexes that need updating

Why is having all overlap values useful?

Consider Incremental Mappers complexity…

Filter function

$$f : X \to Z$$

$(\text{e.g. } O(n^2))$

(needed once)

$+$

Form cover

$$\{U_\alpha\}_{\alpha \in A}$$

$O(nd)$

(needed once)

$+$

Distance Matrix	Cluster Hierarchically	Cut Tree
$D(X_\alpha)$	$\mathcal{C}_H(X_\alpha)$	$\mathcal{C}(X_\alpha)$

$$O(n^2) + O(n^2) + O(n\alpha)$$

(needed per updated level set)

$+$

Form connected components

$$f^*(\mathcal{U})$$

$O(n\alpha)$

(needed every instance of Mapper)

$+$

Form 1-skeleton

$$M(\mathcal{U}, f)^{(1)}$$

$O(n^3)$

(needed every instance of Mapper)

or

Form *n*-skeleton

$$M(\mathcal{U}, f)^{(k)}$$

$O(3^{n/3} \times n^2)$

(needed every instance of Mapper)

Demo

- “The purpose of visualization is insight, not pictures.” - Ben Shneiderman
- “As with the setting of manifolds, one should rapidly metabolize the formal definition and progress to drawing pictures” - Robert Ghrist