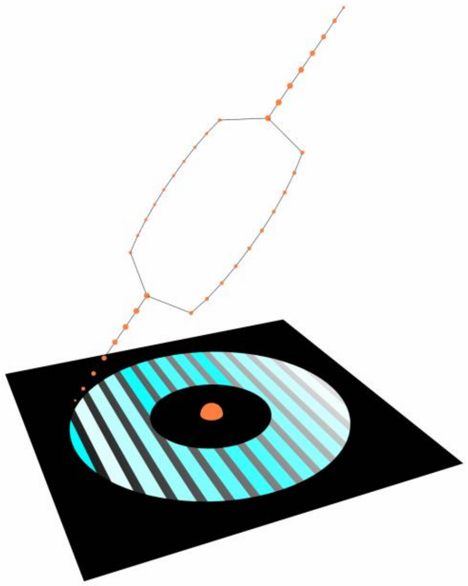
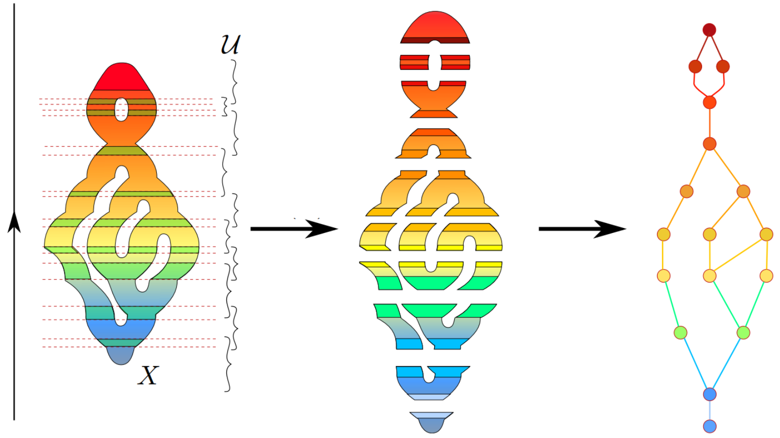
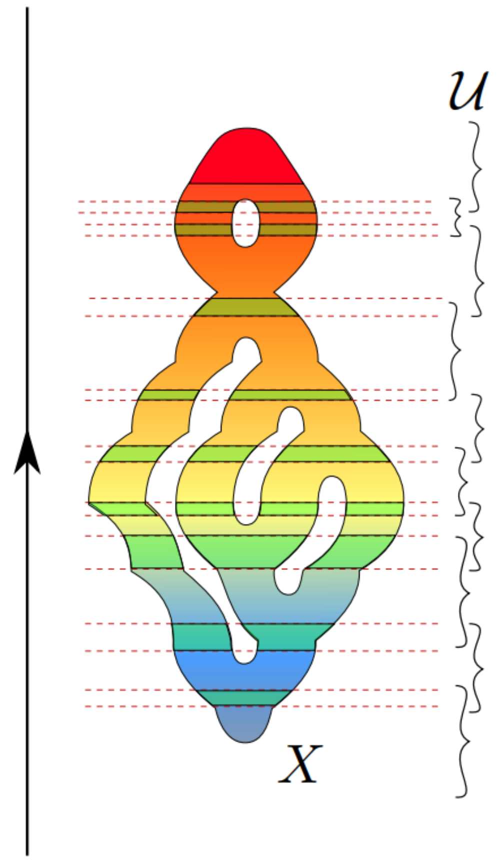


The Mapper Algorithm

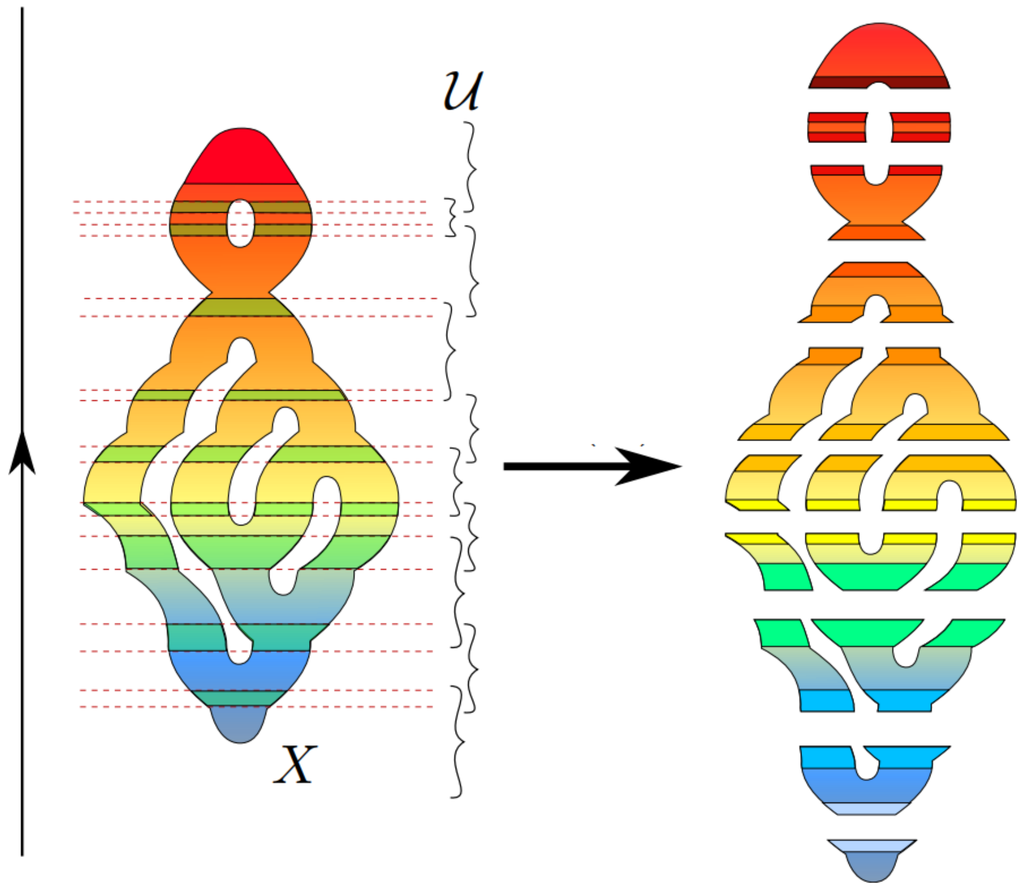


Mustafa Hajij

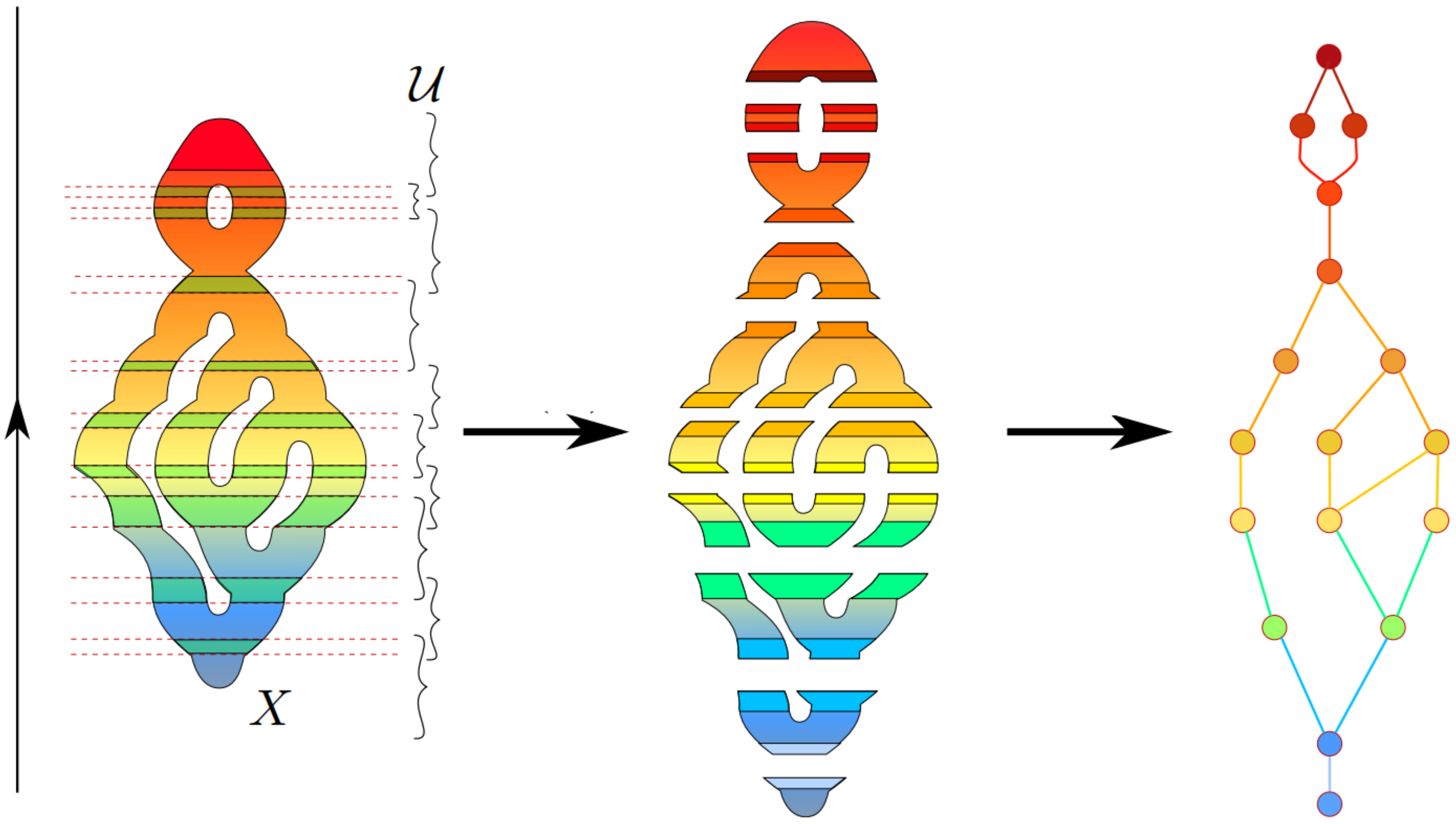
Mapper Construction



Mapper Construction

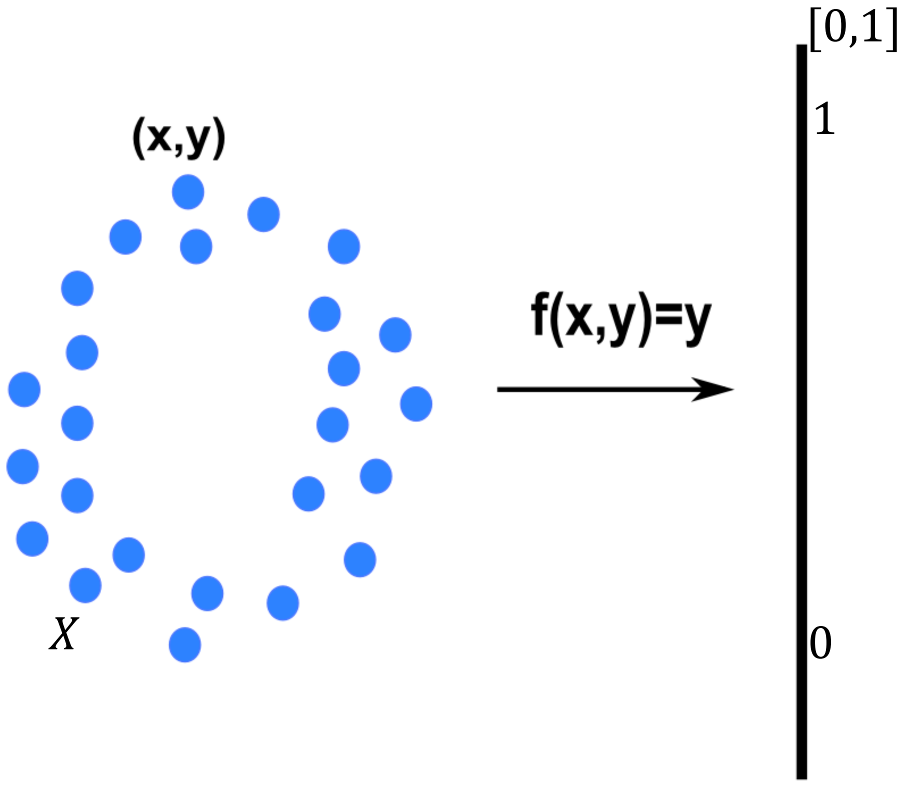


Mapper Construction



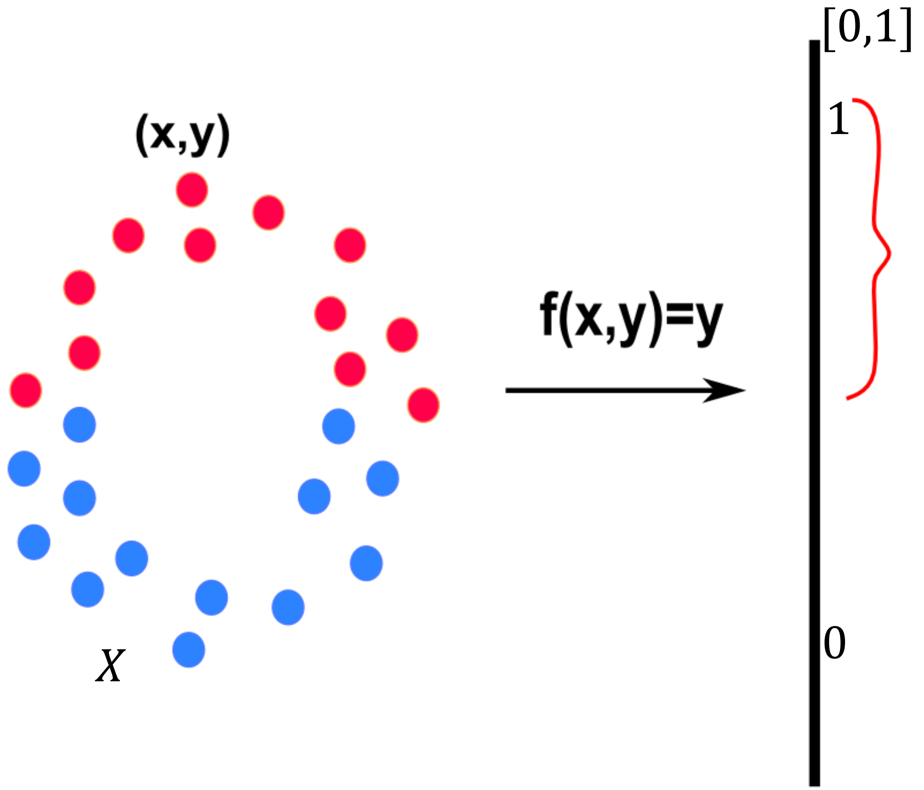
Mapper Example

Suppose that we are given a data set X and a scalar function $f: X \to [a, b]$ defined on every point in X.



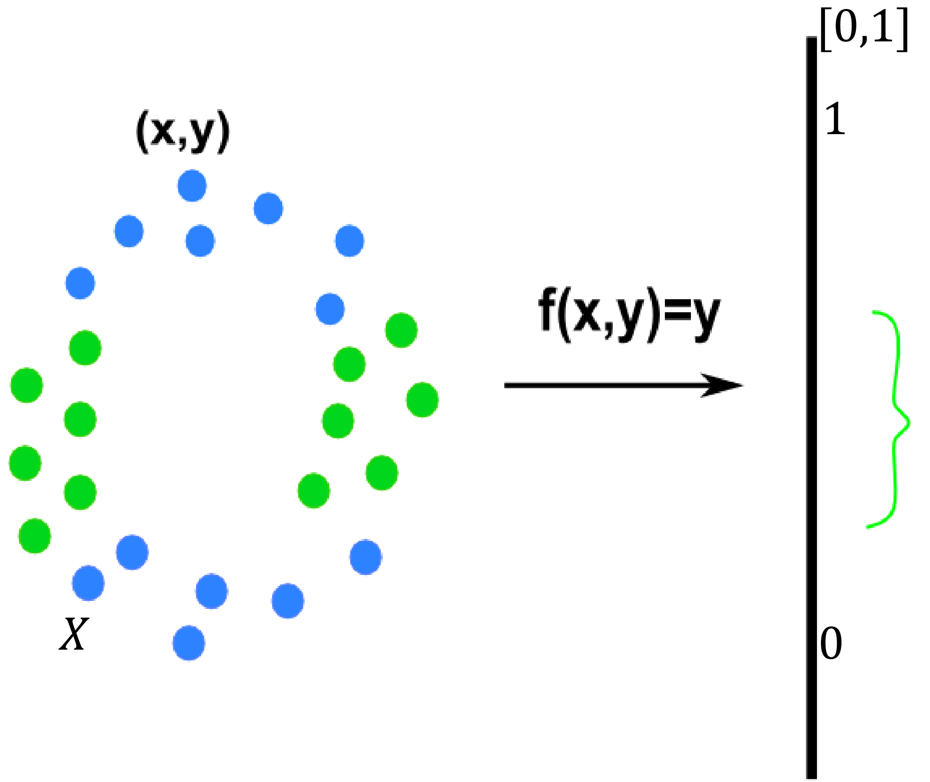
Mapper Example

Suppose that we are given a data set X and a scalar function $f: X \rightarrow [a, b]$ defined on every point in X.



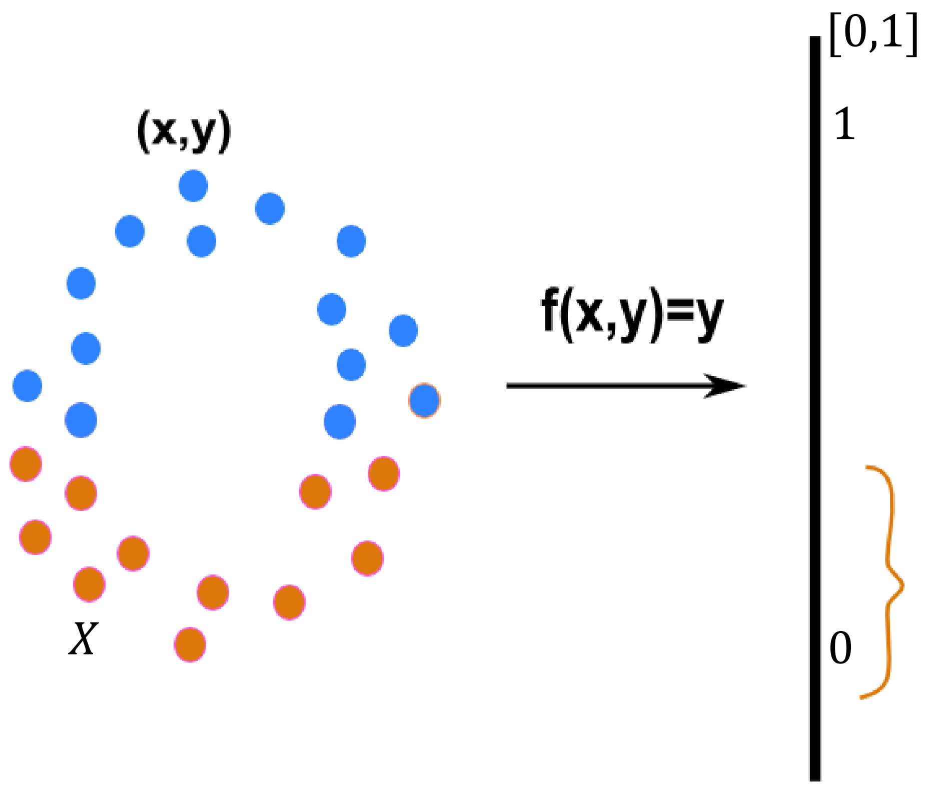
Mapper Example

Suppose that we are given a data set X and a scalar function $f: X \to [a, b]$ defined on every point in X.



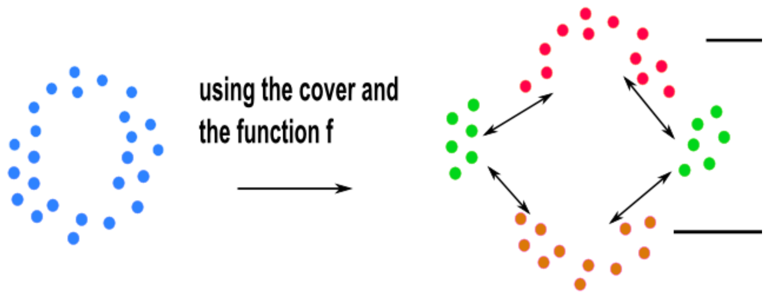
Mapper Example

Suppose that we are given a data set X and a scalar function $f: X \to [a, b]$ defined on every point in X.

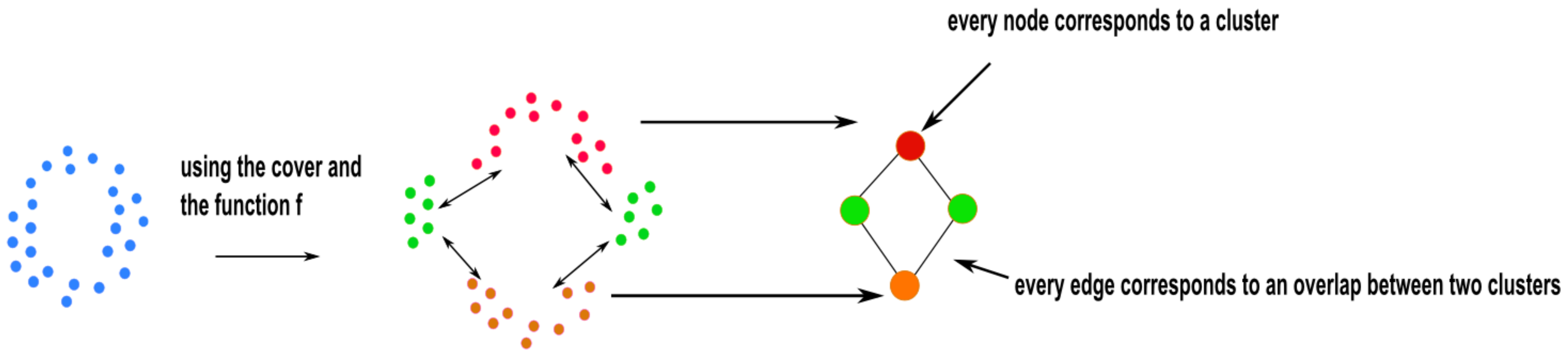


Mapper Example

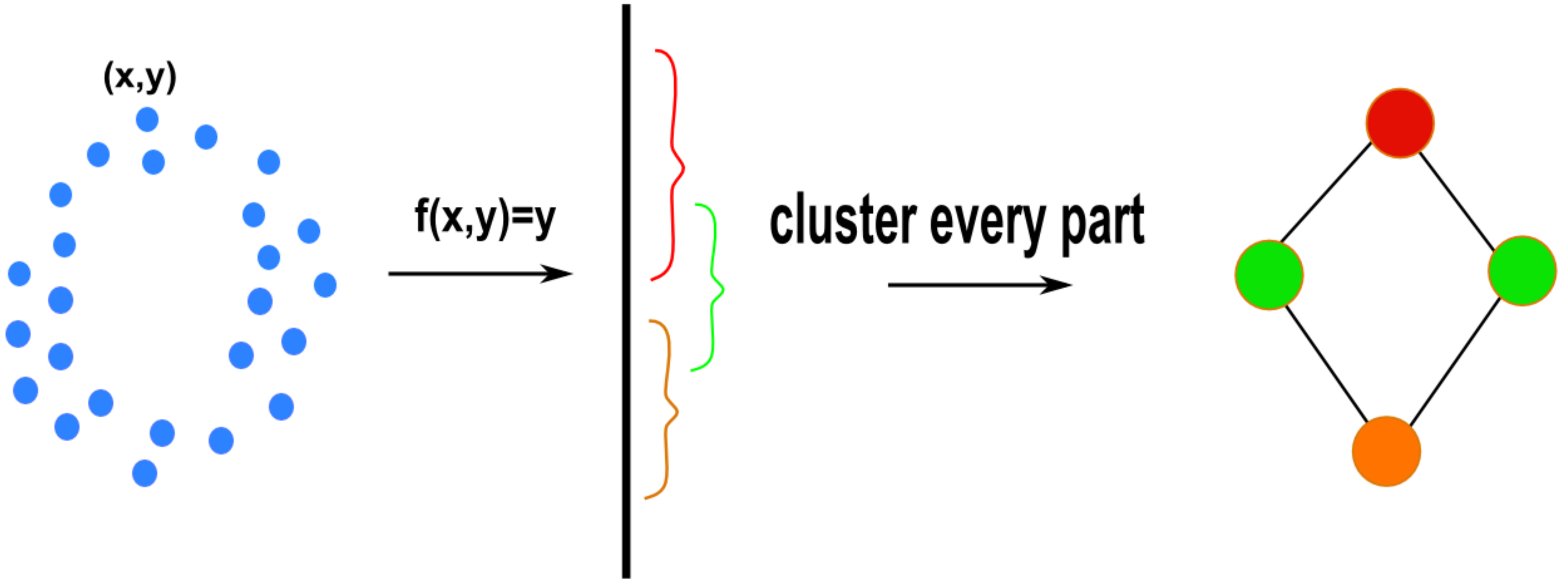
Mapper Example



Mapper Example

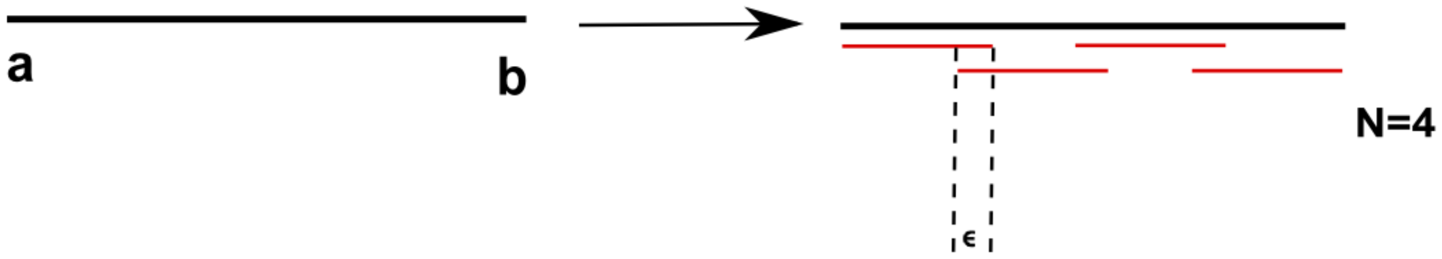


Mapper Example



Cover construction algorithm

Given an interval [a,b] we want to divide [a,b] into N segments each two adjacent ones overlap by an amount **ϵ.**



For example if [a,b]=[0,1], N=4 and ϵ=0.1 then :

The cover consists of the following intervals : [0,0.25+0.05],
[0.25-0.05,0.5+0.05],
[0.5-0.05,0.75+0.05],
and [0.75-0.05,1]

The Mapper Algorithm

Suppose that we are given a data set X and a scalar function $f: X \rightarrow [a, b]$ defined on every point in X.

The Mapper Algorithm

Suppose that we are given a data set X and a scalar function $f: X \rightarrow [a, b]$ defined on every point in X.

1- Define a cover for the interval [a,b]. Say that this cover is $U = \{U_1, \dots, U_n\}$

The Mapper Algorithm

Suppose that we are given a data set X and a scalar function $f: X \rightarrow [a, b]$ defined on every point in X.

1- Define a cover for the interval [a,b]. Say that this cover is $U = \{U_1, \ldots, U_n\}$
2- Consider all the points x in X with f(x) in U_1. Put those points in container, say V_1

The Mapper Algorithm

Suppose that we are given a data set X and a scalar function $f: X \to [a, b]$ defined on every point in X.

1- Define a cover for the interval [a,b]. Say that this cover is $U = \{U_1, \dots, U_n\}$
2- Consider all the points x in X with f(x) in U_1. Put those points in container, say V_1
3-Consider all the points x in X with f(x) in U_2. Put those points in container, say V_2

The Mapper Algorithm

Suppose that we are given a data set X and a scalar function $f: X \rightarrow [a, b]$ defined on every point in X.

1- Define a cover for the interval [a,b]. Say that this cover is $U = \{U_1, \ldots, U_n\}$
2- Consider all the points x in X with f(x) in U_1. Put those points in container, say V_1
3-Consider all the points x in X with f(x) in U_2. Put those points in container, say V_2
4-Do that for every interval $U_i \ in \ U$

The Mapper Algorithm

Suppose that we are given a data set X and a scalar function $f: X \rightarrow [a, b]$ defined on every point in X.

1- Define a cover for the interval [a,b]. Say that this cover is $U = \{U_1, \dots, U_n\}$
2- Consider all the points x in X with f(x) in U_1. Put those points in container, say V_1
3-Consider all the points x in X with f(x) in U_2. Put those points in container, say V_2
4-Do that for every interval $U_i \ in \ U$
5-Run a clustering algorithm on on V_1 and store those clusters.

The Mapper Algorithm

Suppose that we are given a data set X and a scalar function $f: X \rightarrow [a, b]$ defined on every point in X.

1- Define a cover for the interval [a,b]. Say that this cover is $U = \{U_1, \dots, U_n\}$
2- Consider all the points x in X with f(x) in U_1. Put those points in container, say V_1
3-Consider all the points x in X with f(x) in U_2. Put those points in container, say V_2
4-Do that for every interval $U_i\ in\ U$
5-Run a clustering algorithm on on V_1 and store those clusters.
6- Run the same clustering algorithm on every V_i

The Mapper Algorithm

Suppose that we are given a data set X and a scalar function $f: X \to [a, b]$ defined on every point in X.

1- Define a cover for the interval [a,b]. Say that this cover is $U = \{U_1, \dots, U_n\}$
2- Consider all the points x in X with f(x) in U_1. Put those points in container, say V_1
3-Consider all the points x in X with f(x) in U_2. Put those points in container, say V_2
4-Do that for every interval $U_i \; in \; U$
5-Run a clustering algorithm on on V_1 and store those clusters.
6- Run the same clustering algorithm on every V_i
7-Create an empty graph G.

The Mapper Algorithm

Suppose that we are given a data set X and a scalar function $f: X \to [a, b]$ defined on every point in X.

1- Define a cover for the interval [a,b]. Say that this cover is $U = \{U_1, \dots, U_n\}$
2- Consider all the points x in X with f(x) in U_1. Put those points in container, say V_1
3-Consider all the points x in X with f(x) in U_2. Put those points in container, say V_2
4-Do that for every interval $U_i \; in \; U$
5-Run a clustering algorithm on on V_1 and store those clusters.
6- Run the same clustering algorithm on every V_i
7-Create an empty graph G.
8- For every cluster we obtain from $\{V_i | 1 \le i \le n \}$ create a node for the graph G

The Mapper Algorithm

Suppose that we are given a data set X and a scalar function $f: X \to [a, b]$ defined on every point in X.

1- Define a cover for the interval [a,b]. Say that this cover is $U = \{U_1, \dots, U_n\}$
2- Consider all the points x in X with f(x) in U_1. Put those points in container, say V_1
3-Consider all the points x in X with f(x) in U_2. Put those points in container, say V_2
4-Do that for every interval U_i in U
5-Run a clustering algorithm on on V_1 and store those clusters.
6- Run the same clustering algorithm on every V_i
7-Create an empty graph G.
8- For every cluster we obtain from $\{V_i | 1 \leq i \leq n\}$ create a node for the graph G
9- Check overlap between the clusters (nested for loop on all clusters) : whenever there is an overlap insert an edge between the corresponding nodes.

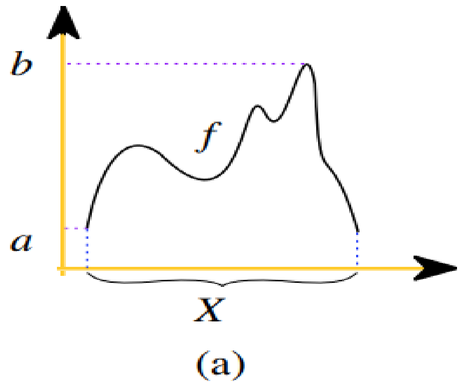
The Mapper Algorithm

Suppose that we are given a data set X and a scalar function $f: X \rightarrow [a, b]$ defined on every point in X.

1- Define a cover for the interval [a,b]. Say that this cover is $U = \{U_1, \dots, U_n\}$
2- Consider all the points x in X with f(x) in U_1. Put those points in container, say V_1
3-Consider all the points x in X with f(x) in U_2. Put those points in container, say V_2
4-Do that for every interval $U_i\ in\ U$
5-Run a clustering algorithm on on V_1 and store those clusters.
6- Run the same clustering algorithm on every V_i
7-Create an empty graph G.
8- For every cluster we obtain from $\{V_i | 1 \leq i \leq n\}$ create a node for the graph G
9- Check overlap between the clusters (nested for loop on all clusters) : whenever there is an overlap insert an edge between the corresponding nodes.
10-return the graph G

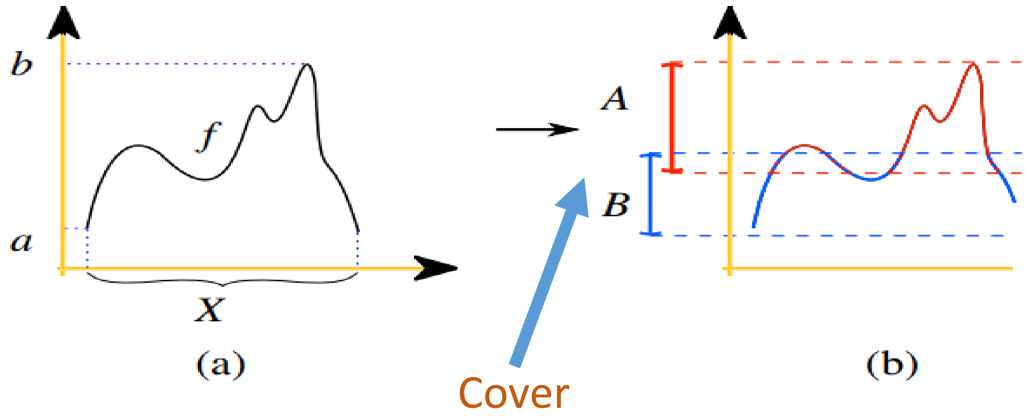
Mapper On Images

The construction of Mapper on a 1d function



(a) A scalar function f : X −→ [a,b].

The construction of Mapper on a 1d function

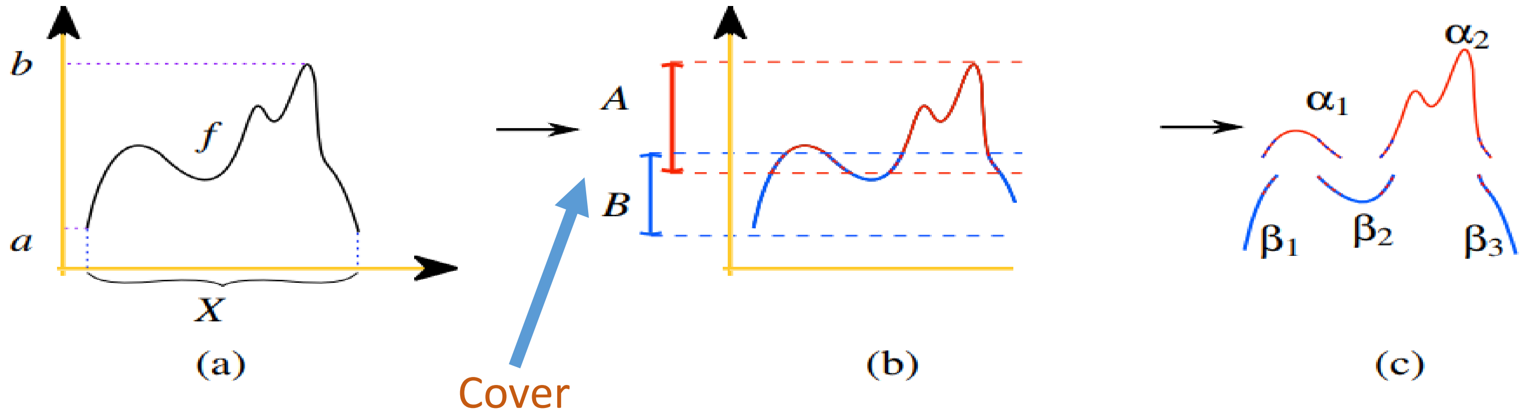


(a) A scalar function f : X –→ [a,b].

(b) The range [a,b] is covered by the two intervals A,B.

I

The construction of Mapper on a 1d function

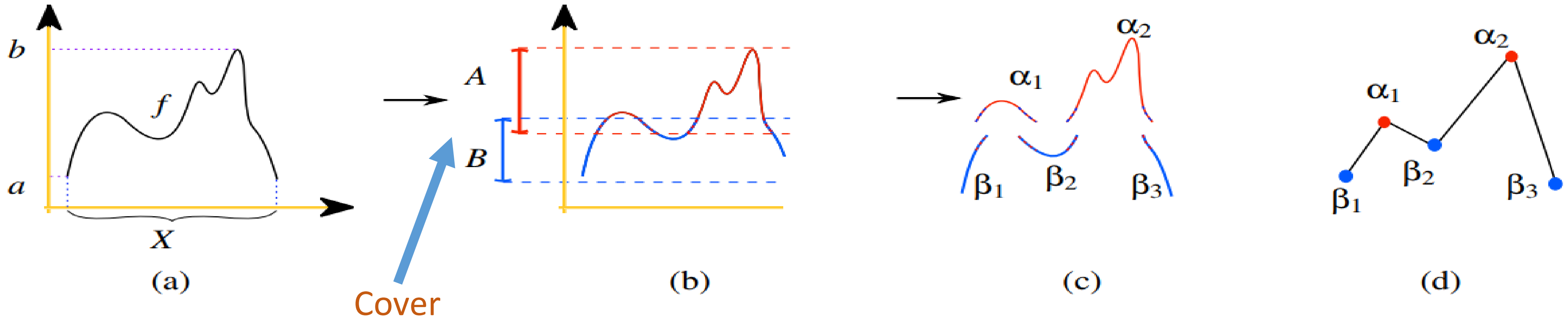


(a) A scalar function f : X −→ [a,b].

(b) The range [a,b] is covered by the two intervals A,B.

(c) This gives a decomposition of the domain the domain X. The inverse image of A consists of two connected components α1 and α2, and the inverse image of B consists of three connected components β1, β3 and β3.

The construction of Mapper on a 1d function



(a) A scalar function f : X –→ [a,b].

(b) The range [a,b] is covered by the two intervals A,B.

(c) This gives a decomposition of the domain the domain X. The inverse image of A consists of two connected components $\alpha 1$ and $\alpha 2$, and the inverse image of B consists of three connected components $\beta 1$, $\beta 3$ and $\beta 3$.

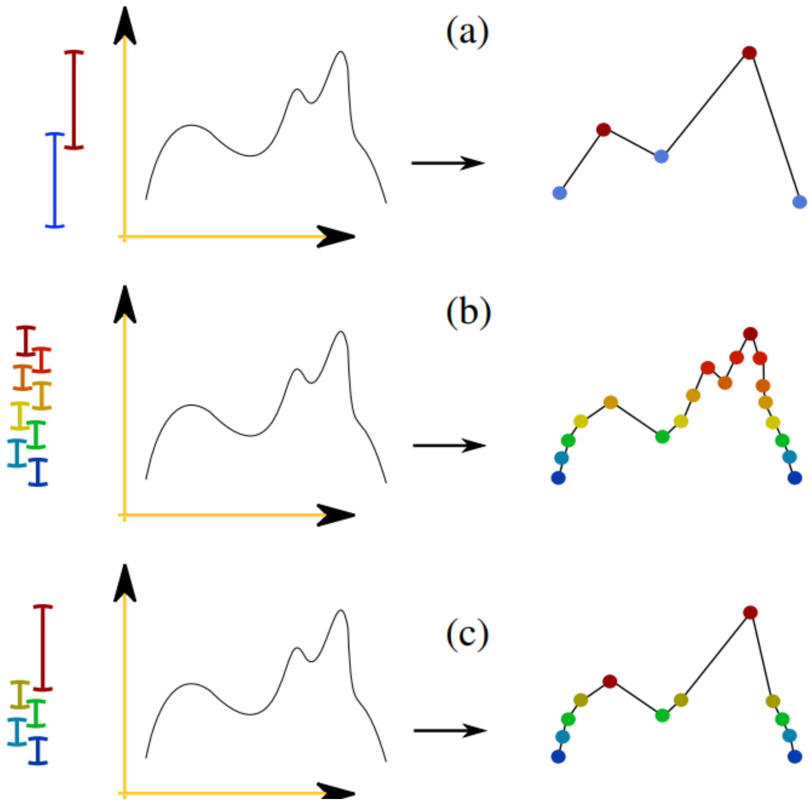
(d) The connected components are represented by the nodes in the Mapper construction.

an edge is inserted whenever two connected components overlap.

Mapper resolution

The construction of mapper depends on the cover chosen for the range [a,b] of the scalar function.

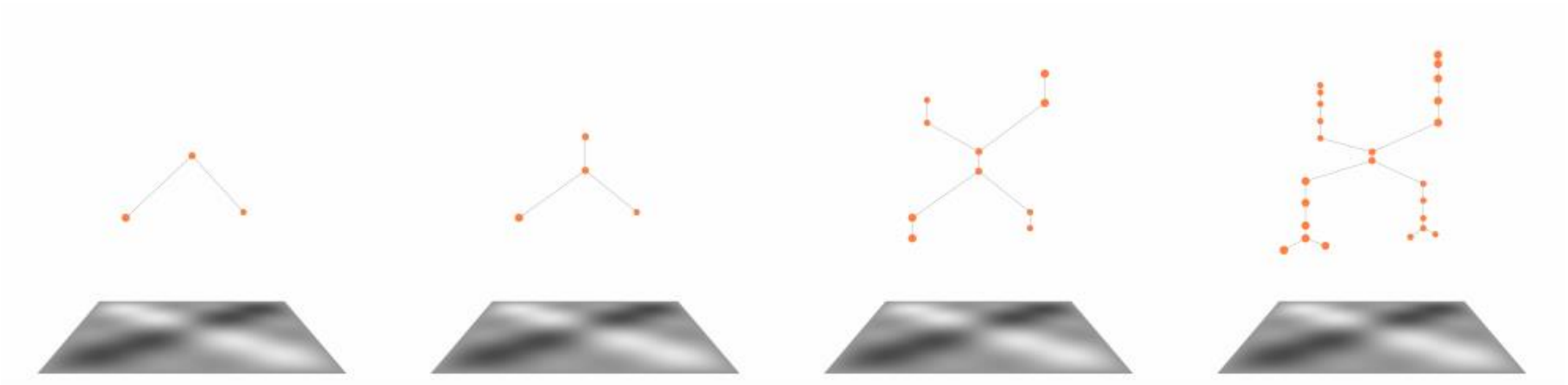
The figure shows three different covers for the range [a,b] and each one gives rise to a different resolution of Mapper.



Mapper resolution

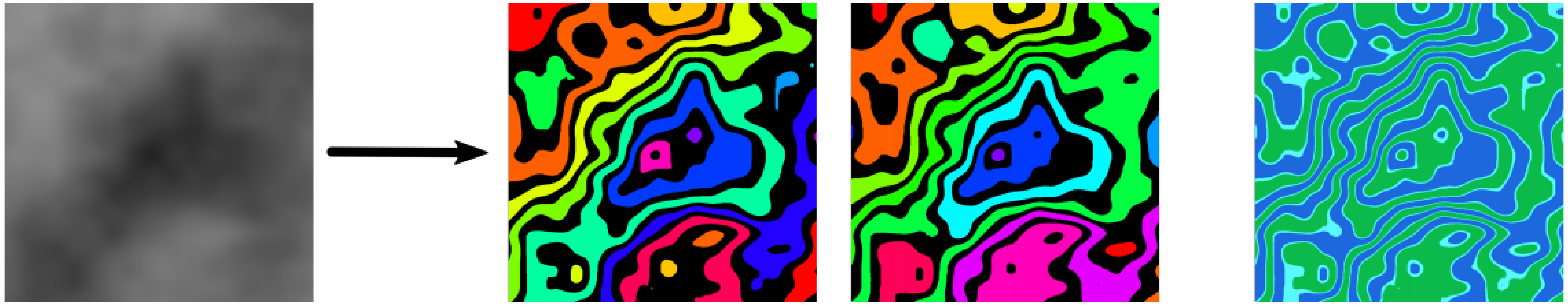
The construction of mapper depends on the cover chosen for the range [a,b] of the scalar function.

The figure shows three different covers for the range [a,b] and each one gives rise to a different resolution of Mapper.



Multi-resolution of Mapper using different cover resolutions. The graphs are constructed from left to right by using 2,4,8,16 slices of the range cover.

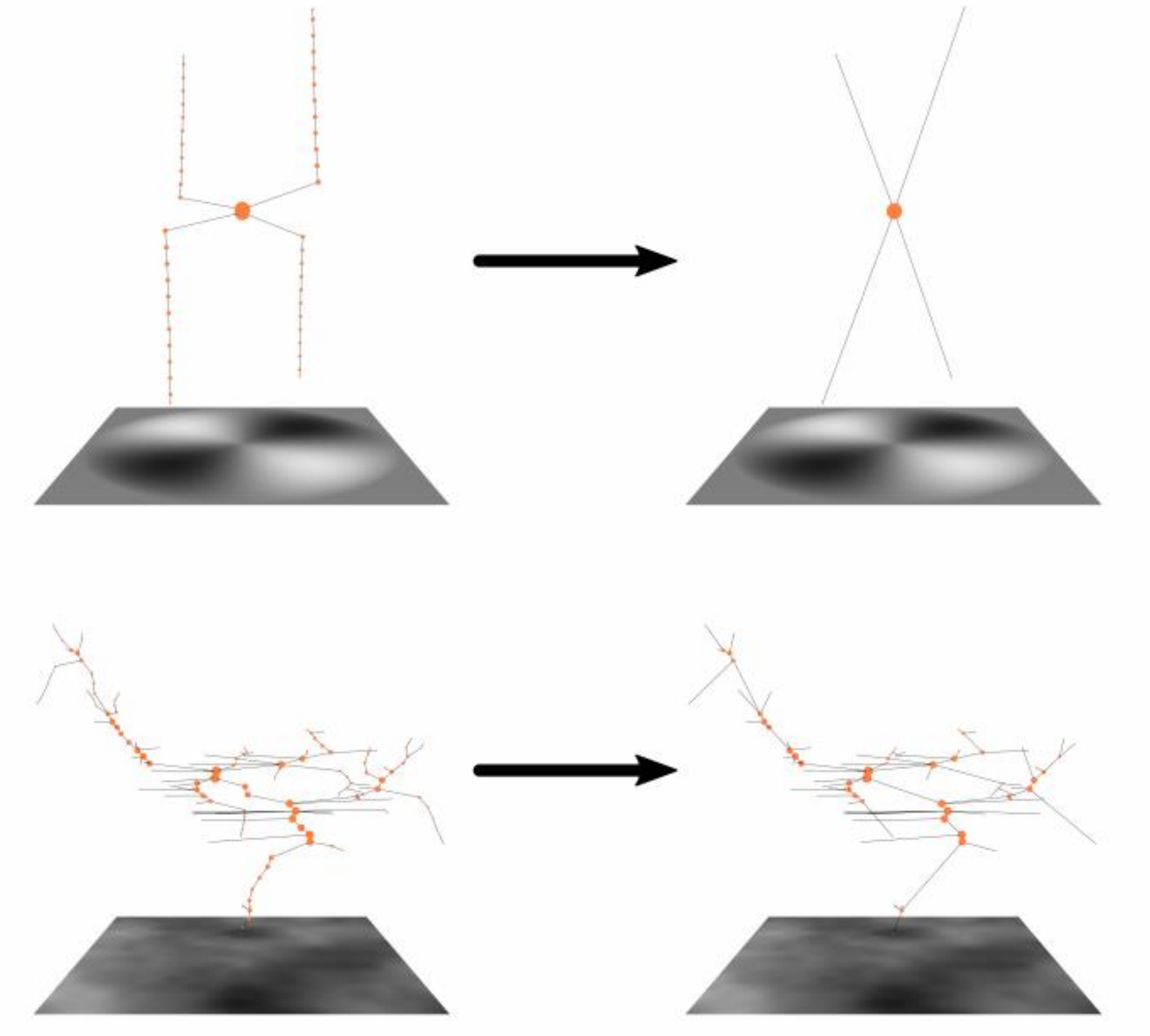
Mapper on images



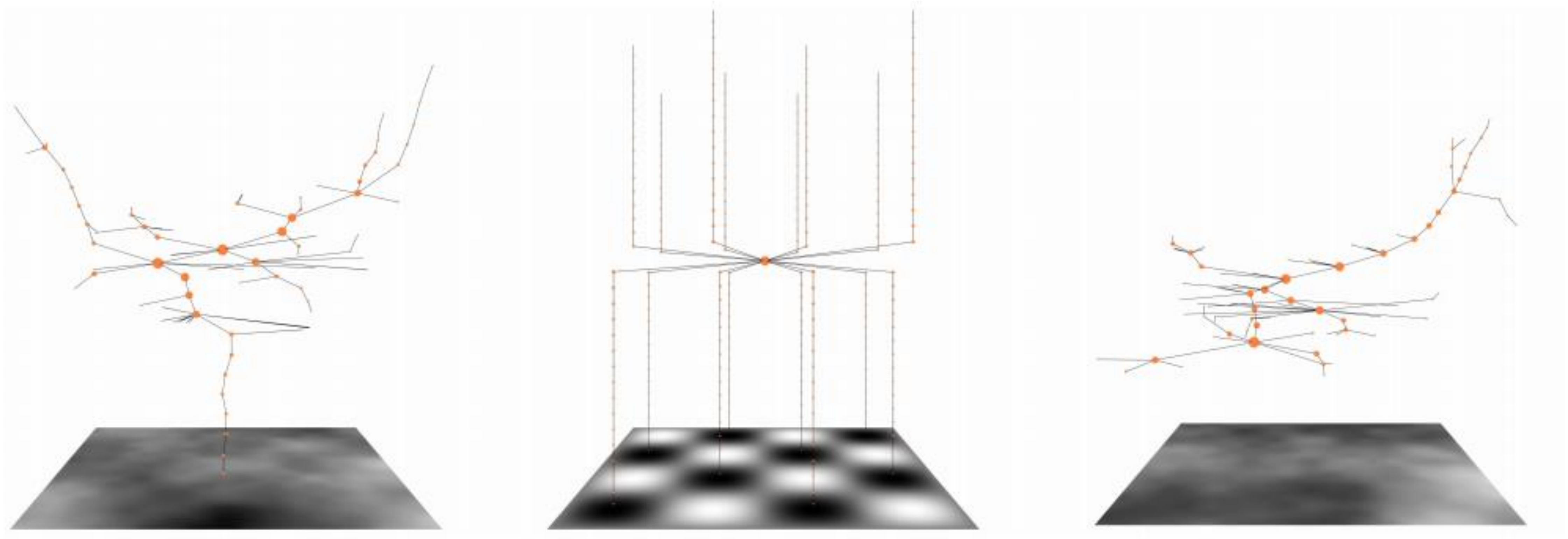
Input Even Odd Overlap

A node in Mapper is a connected component of $f^{-1}((c, d))$, where (c, d) is an open interval in the cover U of the range of f.

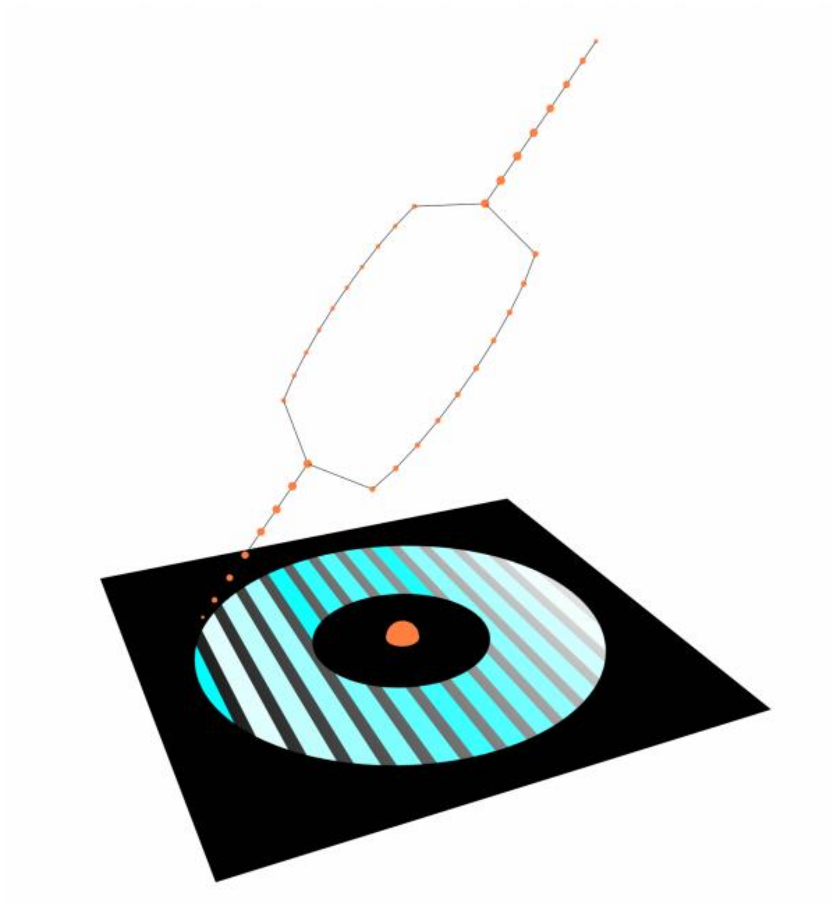
Mapper on images

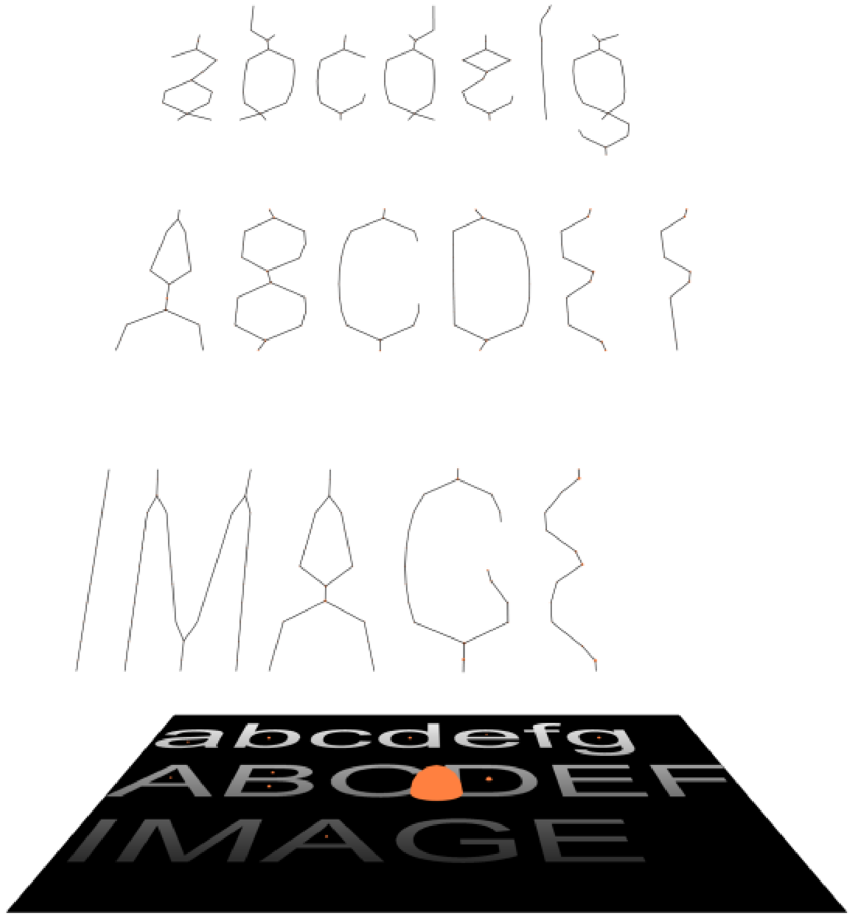


Mapper on images



Shape of an image





Mapper on an image consists of letters. The Mapper construction on this image gives a collection of disjoint graphs that have the same shape of as the letters written in the image.