Stratified Space Learning

Reconstructing Embedded Graphs

Y. Bokor[1]
Mathematical Sciences Institute
Australian National University

EPFL, December 2019

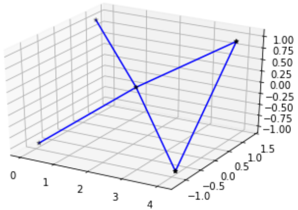
[1]supported by an Australian Government Research Training Program Fee-Offset Scholarship through the Australian National University

Embedded Graphs

We begin by restricting our attention to *graphs*.

Definition

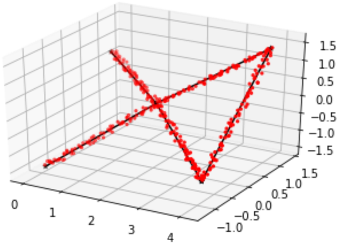
1. An *abstract graph* G consists of two sets: a set of vertices V and a set of edges E.
2. An *embedded graph* $|G|$ in n dimensions is a geometric realisation of an abstract graph.



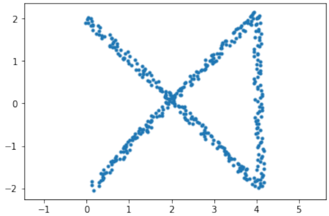
Samples

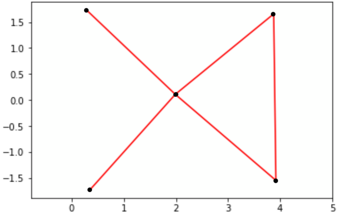
Definition

Given an embedded graph $|G| \subset \mathbb{R}^n$, a *point cloud* sample P of $|G|$ consists of a finite collection of points in $\mathbb{R}^n$ sampled from $|G|$, potentially with noise. If the Hausdorff distance $d_H(|G|, P) \leq \epsilon$, we say P is an ε-dense sample.



Problem Statement





Problem Statement

This is a semi-parametric problem:

- obtain the abstract structure of the graph,
- obtain numerical estimates for the embedding of the abstract structure.

Problem Statement

Let P be an ε-dense sample of $|G| \subset \mathbb{R}^n$, with $|G|$ satisfying some conditions:

Problem Statement

Let P be an ε-dense sample of $|G| \subset \mathbb{R}^n$, with $|G|$ satisfying some conditions:

1. the distance between vertices is bounded below,

Problem Statement

Let P be an ε-dense sample of $|G| \subset \mathbb{R}^n$, with $|G|$ satisfying some conditions:

1. the distance between vertices is bounded below,
2. the distance between edges that do not share a vertex is bounded below,

Problem Statement

Let P be an ε-dense sample of $|G| \subset \mathbb{R}^n$, with $|G|$ satisfying some conditions:

1. the distance between vertices is bounded below,
2. the distance between edges that do not share a vertex is bounded below,
3. the angle between edges at a vertex is bounded below,

Problem Statement

Let P be an ε-dense sample of $|G| \subset \mathbb{R}^n$, with $|G|$ satisfying some conditions:

1. the distance between vertices is bounded below,
2. the distance between edges that do not share a vertex is bounded below,
3. the angle between edges at a vertex is bounded below,
4. at degree 2 vertices, the angle between the edges is also bounded above.

Obtaining Abstract Structure

1. For each sample p, determine $\dim p$.

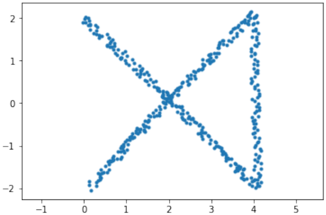
Obtaining Abstract Structure

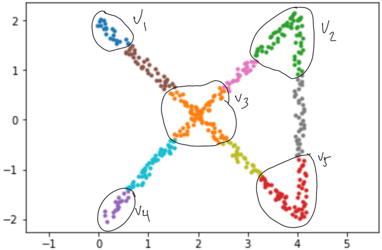
1. For each sample p, determine $\dim p$.
2. Find the number of vertices and edges by clustering the $\dim 0$ and $\dim 1$ samples.

Obtaining Abstract Structure

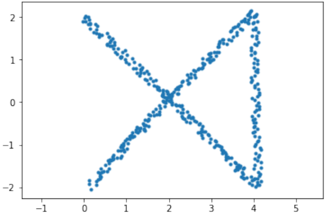
1. For each sample p, determine $\dim p$.
2. Find the number of vertices and edges by clustering the $\dim 0$ and $\dim 1$ samples.
3. Find boundary relations.

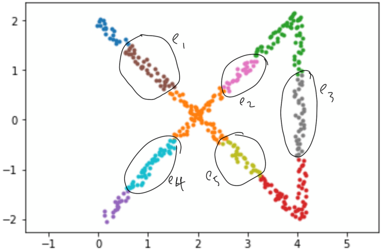
Obtaining Abstract Structure



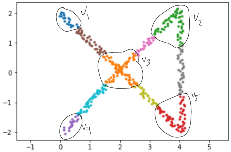


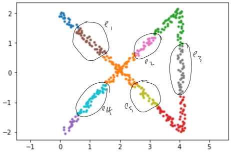
Obtaining Abstract Structure

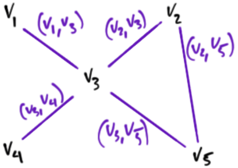




Obtaining Abstract Structure







Obtaining Abstract Structure

Dimension Function

Given a sample q, we consider a ball of radius 10ε centered at q, and look at the samples within this ball. There are several steps to determine if $\dim q$ is 0 or 1.

Obtaining Abstract Structure

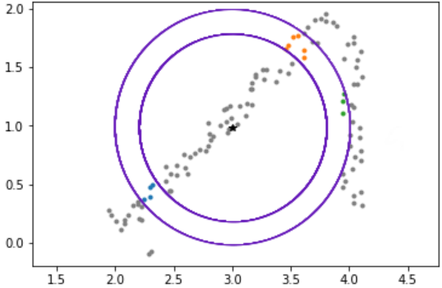
Dimension Function

1. Initialise graph $\mathfrak{G}_q$ with vertices points $p \in P$ such that $d(p, q) \leq 10\epsilon$.
2. For p, p' vertices in $\mathfrak{G}_q$, add an edge between p and p' if $d(p, p') \leq 2\epsilon$.

Obtaining Abstract Structure

Dimension Function

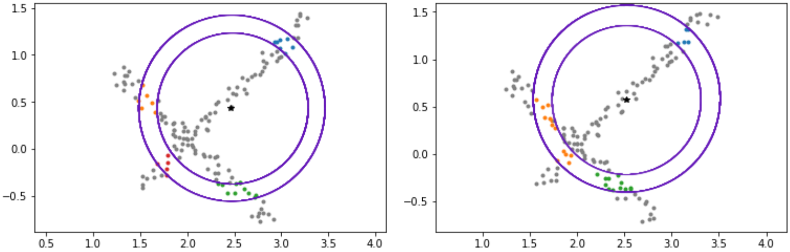
3. If the number of connected components in $\mathfrak{G}_q$ is not 1, **return** dimension 1.



Obtaining Abstract Structure

Dimension Function

4. Else, remove points p with $d(p,q) \leq 8\varepsilon$, and add in edges between $p, p' \in \mathfrak{G}_q$ if $d(p,p') \leq 3\varepsilon$.



Obtaining Abstract Structure

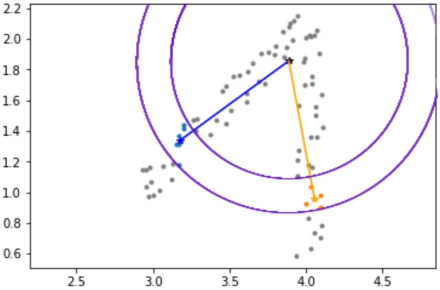
Dimension Function

5. If the number of connected components in $\mathfrak{G}_q$ is not 2, **return** 0.
6. Else, check Angle Condition.

Obtaining Abstract Structure

Angle Condition

7. Find average of coordinates of points in the two connected components.
8. Calculate angle between the line segments from averages to q.



Obtaining Abstract Structure

Angle Condition

9. If angle is less that $2\arccos(1/4)$ **return** 0.
10. Else **return** 1.

Obtaining Abstract Structure

Vertices

1. Initialise empty vertex set V.
2. Initialise graph $\mathfrak{G}$ on $\dim^{-1}(0)$, and connect p, p' if $d(p, p') \leq 9\varepsilon$.
3. For each connected component, add an element to V.
4. **return** V.

Obtaining Abstract Structure

Edges

1. Initialise empty edge set E.
2. Initialise graph $\mathfrak{G}$ on $\dim^{-1}(1)$, and connect p, p' if $d(p, p') \leq 3\varepsilon$.
3. For each connected component, add a unique element to E.
4. **return** E.

Obtaining Abstract Structure

Boundary relations

1. Initalise $|E| \times |V|$ array B of zeros.
2. For each $i \in E$, find points in $\dim^{-1}(0)$ within 3ε of the corresponding points of $\dim^{-1}(1)$.
3. For $i \in E, j \in V$ change $B_{i,j}$ to 1 if samples corresponding to j are within 3ε of samples corresponding to i.

Numerical Description

We now use non-linear least squares regression to best fit the locations of the vertices.

Numerical Description

Partial Objectives

- For $\dim p^{(i)} = 0$: $\phi_i(x_1, \ldots, x_{k_v}, \theta_i) = \|p^{(i)} - x_{j(i)}\|^2$, with $\theta_i = 0$.

Numerical Description

Partial Objectives

- For $\dim p^{(i)} = 0$: $\phi_i(x_1, \ldots, x_{k_v}, \theta_i) = \|p^{(i)} - x_{j(i)}\|^2$, with $\theta_i = 0$.
- For $\dim p^{(i)} = 1$:

$$\phi_i(x_1, \ldots, x_{k_v}, \theta_i) = \|p^{(i)} - \theta_i x_{j_1(i)} - (1 - \theta_i) x_{j_2(i)}\|^2.$$

Numerical Description

Partial Objectives

- For $\dim p^{(i)} = 0$: $\phi_i(x_1, \ldots, x_{k_v}, \theta_i) = \|p^{(i)} - x_{j(i)}\|^2$, with $\theta_i = 0$.
- For $\dim p^{(i)} = 1$:

$$\phi_i(x_1, \ldots, x_{k_v}, \theta_i) = \|p^{(i)} - \theta_i x_{j_1(i)} - (1 - \theta_i) x_{j_2(i)}\|^2.$$

- Combined objective:

$$\Phi(x_1, \ldots, x_{k_v}, \theta_1, \ldots, \theta_n) = \sum_{i=1}^{n} \phi_i(x_1, \ldots, x_{k_v}, \theta_i),$$

with $\theta_i \in [0, 1]$ and $\theta_i = 0$ if $\dim(p^{(i)}) = 0$.

Geometric Correctness

Big question: can we guarantee that the structure we identify is correct?

Geometric Correctness

Big question: can we guarantee that the structure we identify is correct?

YES!!!!!!

Geometric Correctness

Big question: can we guarantee that the structure we identify is correct?

YES!!!!!!

There are a few prpositions which when combined, prove the correctness of our algorithm.

Geometric Correctness

Theorem

Let v be a vertex of $|G| \subset \mathbb{R}^n$, and $p \in P$ a sample. If p is within 3ε of v, then $\dim p = 0$.

Geometric Correctness

Theorem

Let $p \in P$ be a sample which is within ε of edge u, and within 4ε of edge w, u and w having a common vertex v. In addition, assume that the angle α between u and w at v is bounded below by $\frac{\pi}{3}$. Then $d(p, v)$ is bounded above by $2\sqrt{7}\varepsilon$.

Geometric Correctness

Theorem

Let $p \in P$ be a sample which is within ε of edge u, and within 4ε of edge w, u and w having a common vertex v. In addition, assume that the angle α between u and w at v is bounded below by $\frac{\pi}{3}$ and above by $\frac{\pi}{2}$. Then $\dim p = 0$.

Geometric Correctness

Theorem

Let $p \in P$ be a sample with $\dim p = 1$*, which is within ε of edge u, and within 4ε of edge w, u and w having a common vertex v,* $\deg v > 2$*. Then p is more than 3ε away from any sample $\widetilde{p}$ with* $\dim \widetilde{p} = 1$ *and $\widetilde{p}$ more than ε away from u.*

Future Directions

1. Allow for polynomial edges.

Future Directions

1. Allow for polynomial edges.
2. Include higher dimensional strata.

Future Directions

1. Allow for polynomial edges.
2. Include higher dimensional strata.
3. Examine possible uses of machine learning to improve the algorithm.

Future Directions

1. Allow for polynomial edges.
2. Include higher dimensional strata.
3. Examine possible uses of machine learning to improve the algorithm.
4. Repartition sample points with knowledge of the modeled vertex locations.